ÅRGÅNG 3 NR 10 – 28 SEPTEMBER – 25 OKTOBER 1989 – PRIS 27:50 INKL MOMS

AMIGA COMPUTER SOFT

Oberoende Commodore magasin

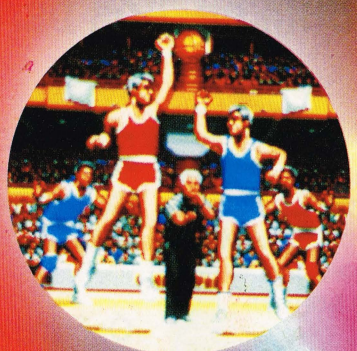
Allt om:

CINEMAWARES MAGISKA VÄRLD

VI TESTAR:
Panasonic skrivare
DOS-2-DOS

KNAPPA IN:
Din egen packer

Massor med teknik och speltester



SKICKA IN OCH VINN!

125-10

HOBBY * DATA * SANDWICHBAR * TEATERBIO * FYRTORNET * LEKSAKER * BARNKLÄDER *

HOBBY * DATA * SANDWICHBAR * TEATERBIO * FYRTORNET * LEKSAKER * BARNKLÄDER * BABYSAKER * TILLBEHÖR * SERVICEVERKSTAD *

VÄLKOMMEN TILL CHIPPEN!

- den fulladdade databutiken, hos barnens bästa i Bromma

AMIGA 500 VÄRLDENS MEST AVANCERADE HEMDATOR
Amiga 500 har 4096 färger, 3-dimensionell grafik animation med bildväxling 50 gånger per sekund och ljud i stereo, hvilket gör Amiga 500 till drömdatorn för Dig som är experimentsugen.

Öppningserbjudande: **4.495:-**

COMMODORE C64
Världens mest köpta hemdator heter Commodore C64. Det finns många goda skäl till framgången, inte minst priset.

Öppningserbjudande: **1.995:-**

CHIPPEN DATABUTIK
B & W I BROMMA
Ulvsundavägen 181
Tel: 08-28 50 20

Commodore

BABYSAKER * TILLBEHÖR * SERVICEVERKSTAD * HOBBY * DATA * TEATERBIO * FYRTORNET *

Ansvarig utgivare:
Klaus Nordfeld

Chefredaktör:
Christian Martensen

Redaktionssekreterare:
Sven Bilén

Redaktion:
Jonny Bergdahl
Pontus Lindberg

Medarbetare:
Sven-Olof Karlsson
Daniel Larsson
Kim Isaksen
Ivan Sölvason
Lars Merland
Sören Kenner (USA)
Hans H. Bang
Henrik Lund
Keith Campbell (UK)
Esben K. Hansen
Amdi Nielsen
Graeme Kidd (UK)
Claus L. Jeppesen
Flemming Steffensen
Rasmus K. Kristiansen
Tom Iversen
Niels Lassen

Redaktionsadress:
Oberoende COMputer
St. Kongensgade 72
1264 Köpenhamn K
Danmark
Tel: 00945 3391 2833
Fax: 00945 3391 0121

Prenumeration:
Postgironummer 74383-1
Tel: 00945 3391 2833
Halvår: 153:-
Helår: 278,50:-

Annonsavdelning:
DSP
St. Kongensgade 72
1264 Köpenhamn K
Tel: 00945 3391 2833
Fax: 00945 3391 0121

Distribution:
Presam

Oberoende COMputer publiceras av koncernen Förlaget Audio A/S.
Samtliga i tidningen utlistade program är testade och kontrollerade före publicering. Förlaget Audio innehar rätten till i tidningen tryckta program och rätt att publicera dem på andra lagringsmedia.
Elektroniska konstruktioner som visas i Oberoende COMputer är kontrollerade, och fungerar enligt konstruktörens och redaktionens fasta förvissning om anvisningarna i tidningen följs till punkt och prickt.
Förlaget Audio påtar sig inget ansvar för följder av tryckfel eller fel i ritningar.
Artiklar eller utdrag därav samt bilder i Oberoende COMputer må ej utan förlagets tillåtelse användas i reklam eller annan försäljningsverksamhet.

Oberoende COMputer
Commodore magasin

Reportage:

Cinemawares magiska värld
Vår reporter har varit i soliga Californien och hälsat på ett av de mest spännande softwarehusen - Cinemaware. Förutom ett spännande reportage om firman kan han också avslöja ett par spelnyheter _____ 30

Teknik:

1541 Tips & Trix
Allt som är värt att veta om Commodores diskettstation. Och nu är det dags att gå in på formatteringsrutinerna _____ 14

CLI-skolan, klass 3
Du behöver CLI! Den här gången får du den kompletta lösningen på hur du ska göra din egen skräddarsydda CLI-disk! _____ 36

Amiga BASIC
Spaltn som lär dig att utnyttja Amigans BASIC optimalt. Den här gången bjuder vi på tre riktigt fräscha programrutiner _____ 34

COMpacker
Efter den långa COMpumon-serien kan vi nu lyfta på slöjan för ett nytt inknappningsprojekt - men denna gången i endast ett (1) avsnitt! Din egen packer! _____ 20

Få stora sprites på C64
Vi avslöjar hur du kan få fram de där riktigt lödiga spritarna på 64:an _____ 22

Maskinkod på 64:an
Dax för maskinkodsserien igen. Du följar väl med? _____ 45

Tester:

Billig och bra trycksak
Vi testar Paanasonic KX-1081, en relativt billig skrivare till både C64 och Amiga. Hur står den sig i konkurrensen? _____ 47

Dos-2-Dos
Att kunna läsa PC-format på en standard-Amiga vore ju inte helt fel. Och ett sådant program har nu sett dagens ljus. Vi testar _____ 40

Soft Check
Vårt alltid så friska testlag tar en titt på de senaste i högen till Commodore 64 _____ 52

Amigames
Amigaspelen blir bara fler och fler. Vi laddar givetvis in de senaste och avger vår dom _____ 26

Snabbisar
Fler speltester - fast i kortform _____ 39

Övrigt:

Puls _____ 4
Arcade Action Aid _____ 51
Läsarundersökning _____ 49
COMpost _____ 10
Lär dig att lösa äventyrsspel _____ 18
Super 20 _____ 42

AMIGA

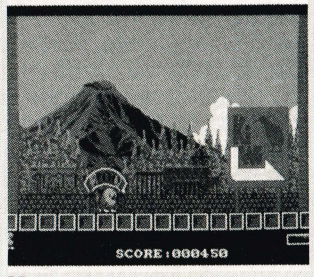
Reportage:

Vi besöker Cinemaware! Sid 30.

Test:

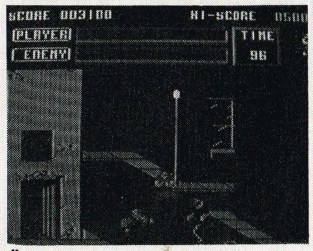
Panasonic KX-1081
- en prisbillig skrivare. Sid 47.

Amigames:



Massor med spel! Sid 26.

Soft Check:



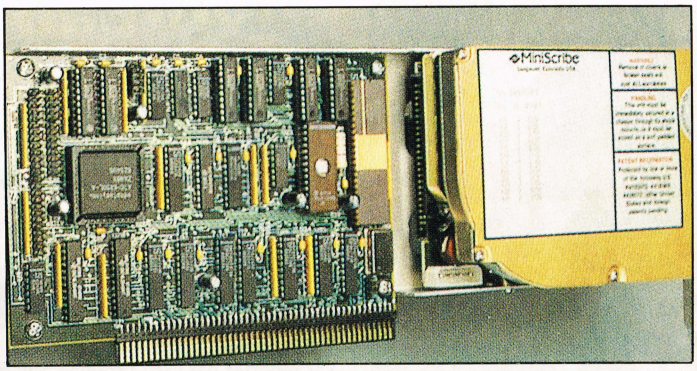
Ännu fler spel?! Sid 52.

Nästa nummer:

26/10

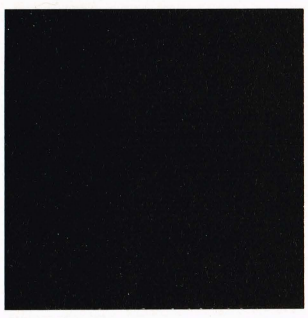
- finns nästa nummer hos din tidningshandlare

PUS



MÅNADENS CODEMASTERS

Senaste nytt från familjen Darling i England är att deras fjorton nästa releaser blir försenade. Varför? Jo, när den äldsta systern klev ur badet ifört sin obligatoriska handduk snubblade hon på Davids kvarglömda rullskridskor och dunkade skallen rätt in i mormors porslinsgrodesamling.
Den gamla blev f-d och kastade en moraklocka efter henne, men missade och råkade träffa hunden som blev så rädd att den hoppade upp och bet av farsgubbens vänstra örsnibb. Som gammal hemvärnsman reagerade han snabbt (han trodde det var ett bakhåll) och kastade ner sin leverpastejmacka i brödrosten varvid det blev kortslutning, strömmen gick och alla datorspelen blev raderade. Vår fotograf fanns på plats och förevigade det hela, men tyvärr hade han glömt sin blixt.



COMMODORES KORT I ÄRMEN

I ett helt år har Commodore i USA legat inne med ett expansionskort som möjliggör anslutning till det mycket spridda PC-nätverket Ethernet. De har dock inte riktigt tänkt offentliggöra det, eftersom de inte omedelbart kunde se några användningmöjligheter för det.

Det var det emellertid andra som kunde när de upptäckte kortet på COMDEX-mässan, så måhända får vi snart se program som kan nätverkskommunikera mellan Amiga och PC.

LÖP UT I SANDEN

I Californien har varje strand med självrespekt minst 20 banor för den speciella sport som heter Beach Volleyball. Nu får du chans att vara med på din 64 utan att få sand i öronen, ity Electronic Arts har gett King of the Beach. I detta spel ska du resa runt på all världens stränder (dock varken Skrea strand, Tylösand eller Pite Havsbad, klar brist) och spela turneringar mot 20 olika motståndare.

Spelet är konstruerat med benäget bistånd av USAs två ledande strandvolleyspelare, så realismen torde knappast fela något.

NY HÅRDDISKCONTROLLER TILL A2000

MicroBotics försöker ta ledningen i kampen om marknadens snabbaste hårddiskcontroller till Amigan med deras HardFrame 2000. Det är ett kort i halvstorlek med plats för en påmonterad vanlig PC-SCSI-hårddisk. Controllern överför data med databusshastighet, vilket betyder att överföringshastigheten (5 Mb/sek) kan köra på i stort sett alla hårddiskar. Samtidigt är DMA-delen mycket snabb i att hoppa på och av bussen, vilket gör den idealisk för multitasking med flera program som ligger på samma hårddisk. Det finns ingen begränsning för diskstorlek, och då den kan styra sju (7!) samtidigt borde den kunna tillfredsställa även en Gigabytehungrande Amigaanvändare.

Den kör Fast File System och kan autoboota om du ligger inne med en Kickstart 1.3

EoA I ROLLSPELSTAGEN

Electronic Arts annonserar nu icke mintre än tre nya rollspels-äventyr till Amiga. Ungefär samtidigt med denna tidning utkommer Swords of Twilight, ett spel där tre personer kan spela samtidigt oberoende av varandra.

I Swords of Twilight finns det åtta parallella världar som är förbundna med en magisk väg. Sju elaka Shadowlords har tagit över sju av världarna, det är endast Albion som fortfarande håller stånd. Du ska hitta sju

INDYS KUSIN?

Firebird kommer nu med sitt första släpp sedan MicroProse tog över rodret. Spelet heter Rick Dangerous, och hade vi inte samtidigt fått se Indiana Jones från Lucasfilm kunde man trott att det skett någon förväxling. Vi stannar vid att tro att Rick är Indys kusin, lika som bär är de i varje fall.

Ja, figurerna är lika och spelidén är densamma, men själva spelen är inte direkt lika varandra. Rickspelet påminner mer om Boulderdash till utseendet, alltså en rätt liten figur som springer omkring i en sorts labyrintliknande omgivning.

Rick Dangerous ska ta sig igenom ett Aztektempel, en egyptisk gravkammare och ett fiendefort för att hitta skatten. Tii sin hjälp har han en pistol, dynamit och... nej, inte en piska, Rick har en stav som han kan slåss med.

Det roliga med spelet är att hela förhistorien kommer i form av en snygg och proffsig liten serietidning, och spelet startar där serien slutar. Kul och snyggt utförd idé.

PSYGNOSIS FÅR SPEL-DIARRÉ

Psygnosis försöker nu att slå sitt eget utgivningsrekord genom att ha icke mindre än 43 nya releaser på tapeten under resten av 1989. Det rör sig dels om konverteringar till andra format, såsom Blood Money och Ballistix till C64, men även om helt nya saker, häribland Infestation, Carthage, Killing Game Show, Aquaventura med flera.

förtrollade svärd i de andra sju världarna och flytta dem tillbaks till sina ursprungliga stenar.

I oktober kommer Keef the Thief, ett mer humoristiskt rollspel. Keef är en tämligen misslyckad figur som egentligen bara är bra på en sak - att stjäla! Så småningom får han höra sagan om Telloc, gudakungen, som en gång härskade över staden med hjälp av en magisk staty. När han störtades slogs statyn sönder i sex delar som spreds ut över kungariket. Den som sätter ihop statyn igen ska bli den nya gudakungen, och Keef är inte sen att anta utmaningen.

Keef innehåller även arkadliknande stridsscener där spelaren ser det hela genom springan i Keefs visir. Vad Keef ska göra väljer du via menyer, någon textinskrivning behövs inte.

I slutet på oktober kommer så The Hound of Shadow, som påstås vara en realistisk simulation av en imaginär värld(!). Det intressanta med spelet är dock att spelet introducerar ett nytt system vid namn Timeline. När du tillverkar en person via Timeline ger du denne en detaljerad bakgrund, ålder, kön, yrke, kunskaper mm. Meningen är att du sedan ska kunna använda din figur i vilket Timelinebaserat spel som helst och att de kunskaper mm du tillägnat dig i ett tidigare spel följer med till nästa.

Låter intressant!

GODIS FÖR COMMODORE-ENTUSIASTER

Är du helt helt galen i Commodore och önskar göra din omvärld uppmärksam på detta, kan vi upplysa att man i England kan rekvirera en katalog över Commodore T-Shirts, paraplyer, nyckelringar, nallar, kulspetspennor och massor med annat.

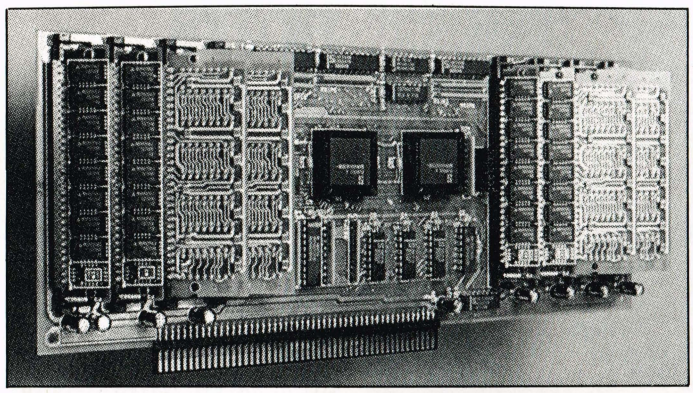
Intresserede kan skriva efter Commodores gåvobroschyr till adressen:
JG Promotions & Marketing Ltd.
8 Brendon Street
Londom W1H 5HE

PULS

SERIELLT MONSTER-SPEL

Activision söker framgång med en konvertering av Segaspelet Altered Beast, där du spelar en romersk Centurion som hämtats upp från graven av Zeus (de har visst lite knepigt med att kunna skilja på greker och romare därborta). För att göra honom lite mer slagkraftig har Zeus trollat lite så att soldaten förvandlas till ett eller annat monster varje gång månen stiger upp. Vilket monster han förvandlas till beror på hur många "spirit balls" du samlat. Du startar som varulv, men kan senare bli vardrake, varbjörn, vartiger, varuhus eller liknande. Var och en av dem har sin egen lilla specialitet såsom drakens eld eller björnens extremt dåliga andedräkt.

Spelet bjuder på 5 levels med var sina megastora sprites, så håll andan tills oktober då spelet utkommer till både Amiga och C64.



BILLIG RAMEX-PANSION TILL AMIGA 2000

En nyhet till för alla RAM-hungrande Amiga 2000-ägare har dykt upp från den amerikanska hårdvarufirman MicroBotics. De har utvecklat ett RAMboard vid namn 8-UP som kan hålla upp till 8 MB dynamiskt RAM och som via en speciell refresh-teknik kör med zero waitstate (det ger högre hastighet).

RAMboardet har dessutom en osedvanligt låg strömförbrukning. Med 8 MB installerat drar det inte mer än 0.8 A, vilket bara är ca 20% av vad en slot normalt kräver.

Med 8-UP kan du välja mellan att köpa MicroBotics SIMUL modulsocklar med RAM installerat eller köpa s k PopSIMMs där du själv ska stoppa i RAM-kretsarna.

För att installera kortet med 8 MB RAM behöver du 8 PopSIMMs, som betingar ett pris på ca 25$ styck.. Själva 8-UP boardet kostar omkring 200$, så med de dalande RAM-priserna är MicroBotics erbjudande faktiskt inte så dumt.

Intresserade kan hänvända sig till denna adress:
MicroBotics, INC.
811 Alpha Drive, Suite 335
Richardson
Texas 75081

BRÄNN ORD-BÖCKERNA

"Jag vill ha 2 stekta mä brö och mycke senapåketchup" säger du till korvnasaren vid Eiffeltornets fot, och han ser frågande på dig tills den lilla dator du har i handen repeterar det hela på perfekt franska.

Framtidsfantasi? Nej, faktiskt inte. Apparaten existerar redan och bygger på taligenkännelse av 35000 inprogrammerade meningar som kan användas av turister. Tills vidare fås den med översättning från engelska till franska, spanska och japanska, men APT har planer på flera andra versioner, som t ex kan användas som möteskalender där du bara talar in dina avtalade möten och som dessutom håller reda på checkkontot (fast är du gift har förmodligen din fru tagit hand om denna funktion redan).

Hela härligheten väger 3 kg och kostar ynka 14.000 kr, så den blir ju snart var mans egendom.
Advanced Products & Technologies
15444 NE 95th Street
Redmond
Washington 98052
USA
Tel: 0091 206 883 8297

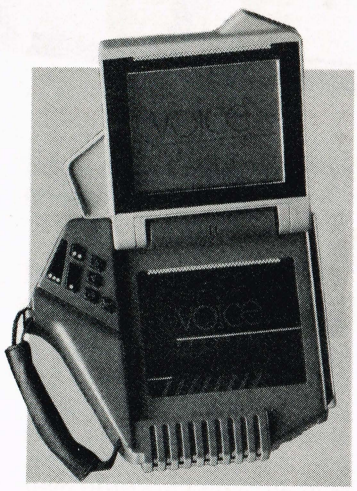
NY ADVENTURE CREATOR

Det har varit snålt med äventyrsdesignprogram till Amiga hittills, men det håller Microdeal på att råda bot på med deras grafiska adventure creator Talespin.

I Talespin kan du använda IFF-filer, vilket gör att du kan använda dina favoritritprogram (t ex Deluxe Paint eller Photon Paint) för att göra den grafiska delen. Talespin klarar dessutom filformaten Degas och Neo.

Med på köpet får du också ett färdigt äventyr som visar vad som är möjligt att krama ur Talespin.

Den enda uppenbara nackdelen för många äventyrstillverkare in spe, är att Talespin kräver minimum 1 MB minne.



SER FINT UT

Vi har just fått in en hög bilder av Titus senaste spel Knight Force. Här spelar du rollen som The Knight of Thunder och ska rädda... ja, ska vi våga en gissning?... joråd, en prinsessa. Hon har tagits tillfånga av Red Sabbath (känd heavygrupp i öststaterna) och det betyder att du ska kämpa dig igenom 25 banor som befinner sig i fem olika tidsåldrar, från stenålder till nutid.

Lyckligtvis har du ett svärd till din disposition (kors vad originellt) och då det samtidigt finns lite strategielement i spelet hävdar Titus att detta spel är början till en ny era i datorspelens värld. Vilken blygsamhet.

Men bilderna ser flotta ut, så döm själv.

Titan, Unit 4
Stannets
Laindon North Trade Centre
Basildon
Essex SS15 6DJ
England

PÅ GRÖNT UPPDRAG

Tja, man kan inte alltid beskylla softwareproducenterna för att vara direkt snabba inte. Spelet Rainbow Warrior till Amiga kunde i varje fall kommit redan för ett par år sedan, men som det slitna ordspråket säger: Bättre sent än aldrig.

Spelet från MicroProse borde vara ute i butikerna när du läser denna nyhet, och här får du chansen arr leva dig in i det heliga ozonkriget. I praktiken gör du detta genom att planera och utföra en massa miljöaktioner. Du kommer inte i TV och ozonhålen blir inte mindre av det, med gnistan och entusiasmen får du uppleva till fullo, lovar MicroProse.

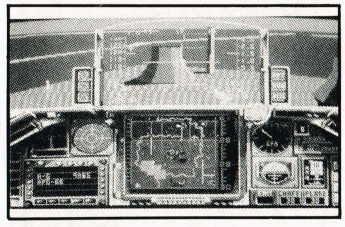
När kommer Greenpeace med ett spel där man via gummibåtar och ankarkättingshängande ska hindra ett atombestyckat fartyg från att gå i hamn?

TIDNINGS-POJKEN PÅ AMIGA

Äntligen annonserar nu Elite att Paperboy är redo att erövra Amigan! Och man har jobbat ordentligt för att få Amigaversionen så lik coin-op-originalet som möjligt. Ljudet har samplats direkt från originalet - i stereo! Och alla figurer du möter i spelmaskinen möter du även i Amigaversionen.

Det finns tre olika svårighetsgrader, och den sista är verkligt jobbig att ta sig igenom. Spelet ska precis ha kommit till affärerna när du läser det här, men vi har fått en version direkt från Elite och alltså redan provat det lite.

Och det är faktiskt riktigt likt coin-upen, dock kunde vi inte erinra oss om ljudet var lika svagt på spelmaskinen som på Amigaversionen. Ett kul spel som du absolut bör ta en titt på!



MOTOROLA 68040

1.2 miljoner transistorer i en enda liten 32 bit mikroprocessor! Som Amigaägare blir det en jobbig väntan innan Commodore får ut tummen och stoppar in denna kraftfulla nyhet i maskinen. Det blir inte i Amigan, utan i en Hewlett-Packard som detta komplexa chip får lov att utföra sina första späda beräkningar.

Motorola 68040 är en vidareutveckling av 68030, vilket betyder att den utan vidare är kompatibel med software till resten av 68000-familjen.

68040 innehåller 5 separata enheter, minnesstyrning, 2 cacheminnen, heltalsprocessor och en matematisk flyttalsprocessor, vilket gör chipet till marknadens snabbaste processor enligt CISC-principen (Complete Instruction Set Computing).

NYA FALCON-UPPDRAG

Om du långt om länge tröttnat på att flyga omkring i din F16 och skjuta ner MIG 25, så kommer det här en nyhet från Spectrum Holobyte till alla innehavare av det utmärkta Falcon. Det har nämligen gett ut en extra mission disk med 12 sprillans nya och farofyllda uppdrag.

Och du kan lugnt slå till efterbrännaren med en gång, för det är inte småsaker du råkar ut för. T ex ska du skydda din bas mot ett stort stridsvagnsangrepp. Uppdragen på disken är dessutom ovanliga på så sätt att de hänger ihop inbördes. Förstör du en bro i ett tidigt uppdrag förblir den förstörd även i de senare, vilket gör det lättare att utföra de senare uppdragen.

Marknadens Bästa Videotexprogram
för Commodore och Amiga

DTEX-64 250:-

Ett avancerat och prisvärt Videotexprogram för Commodore 64/128. Inbyggd svensk teckenuppsättning. Utskrifts-, lagrings & editeringsmöjligheter. För 75/1200& 1200/1200 baud.

TRILOG 1.150:-

Musstyrt och lättarbetat Videotexprogramför Amiga. Automatinloggning och Telefonbok. Utskrifts- & lagringsmöjligheter. För 75/1200 och 1200/1200 Baud.

ÖVRIGA VIDEOTEXPROGRAM

IBM-PC/XT/AT/PS2 + kompatibler
DTEX-80
Mycket avancerat och lättarbetat Videotexprogram. Högupplösande Prestelgrafik. Autoinloggning och telefonbok. Utskrifts, lagrings- & editeringsmöjligheter.
P: 1.166:-

Modem
RPTI TRANSMODEM
Ett mycket litet och behändigt modem (22x14x4 cm), autouppringning, autosvar och 100% Hayeskompatibelt. Indikeringslampor för 8 funktioner. Inbyggd högtalare. 1 års fabriksgaranti.

Finns i två utföranden:
TM-1200A
300 & 1200 Baud, full duplex. **P: 1.347:-.**
TM-2400 A
300/600/1200/2400 Baud, full duplex.
P: 2.700:-

Macintosh
VICOM
Fullständigt kommunikationsprogram för både Videotex och VT 52/VT100. Telefonbok & autoinloggning.
P: 2.300:-

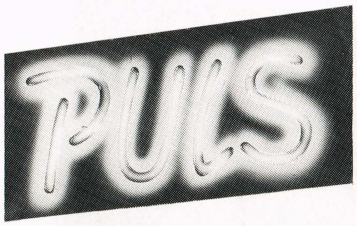
Amstrad CPM
CHIT CHAT
Fullständigt kommunikationsprogram för både Videotex och TTY-kommunikation. Utskrifts- & lagringsmöjligheter.
P: 990:- (seriellt interface **1.290:-**).

Du behöver följande:
RS-232 kabel (PC, Amiga 500/2000 och Atari ST). **P: 77:-**
RS-232 Interface (C64/C128). **P: 394:-**
Seriellt interface för Macintosh. **P: 225:-**
Telefonkabel. **P: 50:-**

Beställ per brev eller telefon. Varorna skickas mot postförskott. Porto och postförskott tillkommer. Moms ingår!

GulaTidningen
Box 6615 113 84 Stockholm 08-33 79 00

Om du söker ett Videotex/kommunikationsprogram vi inte tagit upp - ta kontakt med oss. Troligen kan vi ta fram det till ett mycket förmånligt pris.



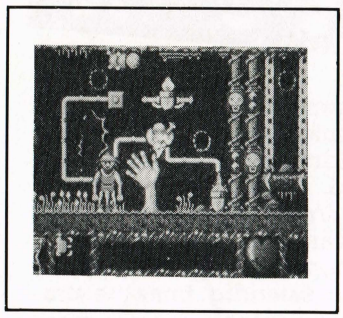
ÄVENTYR-SNYTT

Det surrar massor av rykten kring Infocom och Activision. Enligt vanligtvis välunderrättade källor ska Activision ha uttryckt önskemål om att stoppa samarbetet med populära Infocom på grund av ekonomiska problem.

Samtidigt står det klart att Dave Lebling och Steve Meretzky har lämnat Infocom, enligt ryktena i protest mot Activisions kommersialisering av Infocom, som Dave Lebling bl.a. är en av grundarna av.

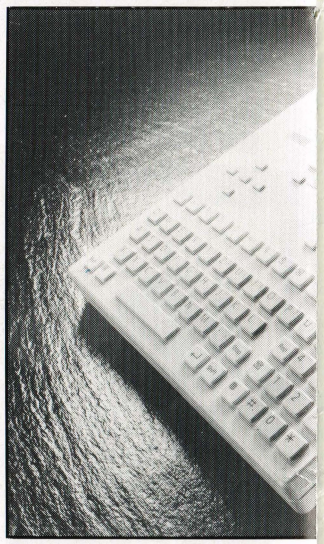
Om de två succépojkarna kommer igen med en ny äventyrsfirma vet vi i skrivande stund inte. Infocom fortsätter dock oförtröttat, nu i nya lokaler på västkusten.

Activision i England meddelar att de fortfarande kommer att distribuera Infocomäventyr i Europa, men det begränsas till Amiga och Atari samt Macintosh och PC.



Rainbirds senaste utsläpp till Amiga handlar om hurudjuren i en stor ek. För varje insekt du smäcker blir ditt bankkonto fetare och fetare, och pengarna kan du givetvis använda till att köpa bättre och mer effektiva bekämpningsmedel. Ett alternativt sätt att skaffa pengar på är via casinot, men här kan ju behållningen gå åt två håll, om man säger.

Verminator förväntas snarast finnas i handeln inom kort.



TELEFON MED AMIGA-HJÄRNA

Om du trodde att det bara var Amigor som tänkte med en Motorola 60000-processor, trodde du fel. Schweiziska Habimat har just lanserat världens mest avancerade telefon, och den är baserad på högst densamma processor.

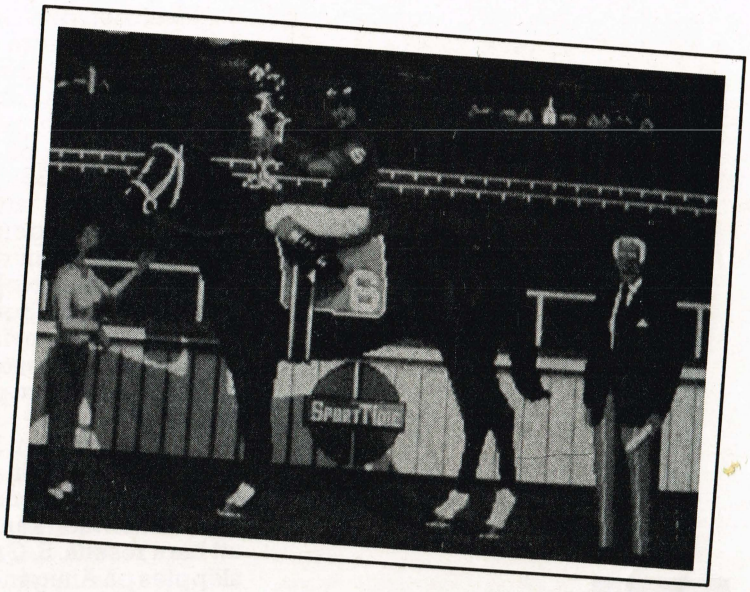
Apparaten har en grafisk display med en upplösning på 64.000 pixels och är ikonstyrd som du kan se på bilden. När du så ska ringa upp t ex en affärsförbindelse trycker du bara dennes första bokstav, så rullar alla som börjar med denna bokstav fram på skärmen (ja, alla som du kodat in, alltså). Här kan du också se olika data om personen, som rabattsatser, födelsedag mm så att du kan smöra på maximalt. Dessutom kan Habimax HT-X erbjuda automatisk återuppringning, möteskalender, stoppur, password och mycket an-

nat. Sist men inte minst kan man faktiskt tala i den, dvs använda den som telefon, men för det nöjet får man betala ca 12.000 för 44 kB-versionen och väsentligt mer för 512 kB-varianten.

I SADELN PÅ DIN AMIGA

Istället för att åka till Täby eller Jägers för att tillfredsställa din galopperande spellust, kan du snart parkera dig i bästa fåtöljen och göra dina insatser i Mindscapes galoppspel.

Spelet heter Omni Play Horse Racing, och du startar med att inspektera hästarnas form och allmänna hälsotillstånd så du har något att bygga dina satsningar på. Mindscape meddelar att det i senare moduler blir möjligt att själv äga och träna sina hästar och sedan låta dem tävla mot några av tidens berömda kusar.



CHIPPEN - NY DATORBUTIK

En ny datorbutik har sett dagens ljus i Bromma. Det är mannen bakom Svenska Pentagon, Antti Koponen, som tagit initativ till butiken som mycket passande fått namnet Chippen.

Och den är heller inte svår att hitta, den ligger mitt i nyöppnade "Barnens Bästa", ett varuhus för allt som har med barn att göra.

Chippen räknar ochså med, att förutom den normala affärsförsäljningen även anordna seminarier och kurser för datorintresserade, både på nybörjare- och expertnivå.

Antti Koponen räknar själv med att det hela ska bli en succé, och säger: "Chippen ska bli sveriges bästa datorbutik - med tiden!"

CHIPPEN
Barnens Bästa
Ulvsundavägen 181
Bromma

AMIGA FICK KNIVEN AV WP

WordPerfect Corporation, som konverterade detta från början PC-ordbehandlingsprogram till stor glädje för mången Amigaägare, meddelade härförleden att de ville stoppa all vidare produktion på Amigasidan. Då de förlorat $800.000 sedan starten är det också förståeligt.

De höll annars på med en version 6 med grafiskt användarinterface, men efter att Amigastoppet blev känt ringde hundratals Amigaägare till firman och sade att de ville vara fulltnöjda med en uppgradering motsvarande 5.0-versionen på PC. Detta fick direktören Lynn LeBaron att ta det hela under övervägande igen, så nu sitter det igen tre programmerare och arbetar på högtryck på det nya projektet.

Deras andra project, PlanPerfect, ett spreadsheet till Amigan, stod dock inte att rädda, trots att det kommit så långt som till teststadiet. Vi hoppas att de som crackat WP härmed får lite dåligt samvete.

EFFEKTIV VIRUS-KILLER

Anti-virus från Devware har i stort sett allt du kan önska dig av en viruskiller till Amiga. För det första kan de skydda dina egna diskar från infektion. Dessutom kan den under söka bootblock och reparera dem. Och som en nyhet kopierar den också non-standard bootblock från spel till en speciell disk, så att om något av dina spel skulle bli förstört av en virus kopierar du bara tillbaka originalbootblocket till disketten via Anti-Virus.

Programmet letar också efter virus i RAM och klarar även de s k Trojanska hästarna där ett virus smugglas in gömt i ett annat program. Adressen?
Devware
10474 Rancho Carmel Drive
San Diego
California 92128
USA
Tel: 0091 619 673 0759

Grafikprogrammering på Amigan

Hej Oberoende! Jag har köpt boken "Abacus Machine Language Programming". Den är jättebra men det finns för lite exempel om grafik programmering. Kan ni rekomendera en annan bra bok som behandlar detta ämne! Forsätt med Amiga Power Works och Assembler skolan och berätta helst om Coppern, scrollrutiner osv.
"68000-Fan"

Böcker till Amigan blir ett allt roligare kapitel eftersom det ständigt dyker upp nya. "Inside Amiga Graphics" är en bok som bara tar upp ämnet grafik på Amigan. Det är den enda bok jag känner till som bara behandlar ämnet grafik, men "System Programmers Guide" tar ju upp hårdvaruregisterna och då kan man själv experimentera sig fram.

Man kan även lära sig en hel del om grafikbiblioteken i "Programmers handbook 1" och i ROM Kernel böckerna om "Liberaries & Devices" men detta är nog inte vad du söker i första hand.

Fem fräscha frågor

Howdy Oberoende COMputer
1. Vad är en reset?
2. Hur fungerar den?
3. Var köper man den?
4. Vad kostar den?
5. Finns Great Gianna Sisters till Amigan?
Patrick Sjöberg, Sundbyberg

1. Reset betyder "återställning" och en reset på datorn gör just en återställning. Resetrutinen är den första rutin som datorn kör efter en kallstart, dvs då man sätter på datorn. Man kan se den i arbete på en 64:a om man skriver SYS 64738.
2. Rent tekniskt ansluter man GND med RESET någonstans på kortet, lämpligen med en reset knapp.
3. En reset knapp köper man i vilken komponentbutik som helst och löder fast den enligt anvisningar i den utmärkta Imponansserien. Resetknappar finns dessutom på alla utility cartridges.
4. Att göra en egen kostar max en 10 kronor för knappen och några kronor för sladd och lödtenn. Cartridges är ju dyrare men men kan en hel del mer än att bara resetta. 5. Great Giana Sisters släpptes på Amigan men efter en juridisk tvist med tillverkarna av Super Mario Brother, vilka inte tyckte att GGS byggde på någon speciellt originell idé, stämde sin konkurrenter.

Resultatet blev att GGS inte längre var tillåtet att sälja. Trist att det roligaste Amigaspelet skulle försvinna på det viset men det lär vara sant.

Andrahandsvärdet på Amiga

Hej och tack för en bra tidning.
Jag har vidare ett par frågor som kommer här:
1. Kommer alla spel från Amigan ut på 64:an också?
2. Kommer de ut samtidigt eller dröjer det innan de kommer till 64:an?
3. Vad kostar en Amiga och vad måste man mer för tillbehör?
4. Varför är grafik och ljud bättre på en Amiga än på en 64:a?
5. Finns det bara diskspel till Amigan?
6. Hur högt är andrahands värdet på en Amiga? 7. Till vilken dator finns det flest spel?
8. Vilken dator har bäst spel?
9. Tycker ni att Amigan är för dyr att bara spela på?
10. Kan man kopiera Amigaspel av kompisar eller måste man köpa? Pris?
11. Finns det någon dator som totalt sett är bättre än Amigan?
12. När kommer W.G. Hockey till 64:an?
Anonym från Laholm

Jag svarar inte på dina frågor en och en utan tar allt i en gröt. Om man jämför en Amiga och en 64:a så finns det klara skillnader. Amigan kan hantera grafik och ljud på ett mycket snabbare och smidigare sätt än en 64:a.

Amigabildernas upplösningen, d.v.s. hur många punkter man kan påverka från datorn, är betydligt fler än på en 64:a vilket gör att kurvor blir mjukare och att bilder kan vara snyggare. Det är två datorer som talar helt olika språk. Därför blir det till att skriva ett helt nytt spel om man skall konvertera dem mellan maskinerna vilket gör att utgivnings datum skiljer på spelen.

Vissa spel kräver helt enkelt en mera avancerad dator, varför en 64:a ibland inte håller måttet. Ta t.ex. Populous som inte kommer att dyka upp till 64:an på grund av att det är så komplext att det inte går att trycka ihopa i en 64:a. Samma fenomen gäller Dragon's Lair.

Den gamla 64 versionen är inte speciellt lik den Amigan visar upp. Att kopiera spel är lika olagligt oavsett formatet men fullt möjligt oavsett formatet.

Till Amigan finns lyckligtvis inga band, utom det tapestreamerband som knappast är aktuella att ta upp för att besvara din fråga. Amigan är den bästa datorn man kan köpa för en rimlig summa pengar.

Jag tycker att en MAC II är lika rolig eller roligare att använda än en Amiga, men min plånbok bleknade då jag fick de vad den kostade. Beträffade andrahandsvärde och andra priser osv. kan du ju titta i annonser.

Kringutrustning och andra tillbehör har presenterats i en tvådelad artikel i denna tidnings två föregående nummer. W.G. är ute i handeln nu!

Anropa maskinkod från BASIC

Hej! 1. Jag undrar varför ni inte har någon maskinkod till Amigan. Finns det någon bra bok (som man förstår) om maskinkod till Amigan?
2. Varför säljs det inte reset knappar längre?
4. Finns det någon bra rutin som anropar maskinkod från BASIC?

Hejdå från "Nybörjaren"

1. Vi har visst en serie om maskinkod till Amigan som heter Assemblerskolan. Den finns kanske inte alltid i alla nummer men dyker upp med jämna mellanrum.
2. Bra böcker som är enkla om maskinkod på Amigan - Tja, man kan välja Programmera 68000. Detta är en

svensk, bra och ganska lättförståelig bok. Problemet med den är att den inte är specialicerad på Amigan.

Alla exempel arbetar med CP/M 68K eller Unix. Inget av dem är speciellt likt Amigans operativsystem. Av engelska böcker är "Abacus Amiga Machine Language" och "Assembly Language Programming on the Amiga" två bra böcker. Vilken man väljer spelar stäng taget ingen roll för de är båda mycket bra. Problemet är ju att ett klurigt ämne blir ännu klurigare om man dessutom skall tolka engelskan.

3. De säljs visst! Kolla på alla cartridges. En reset kan vara "unstopable" och inte. En vanlig resetknapp är inte "unstopable", men en som cartridges genererar går inte att stoppa. Programmen tar hand om en vanlig reset men en cartridgereset kan inget program skydda sig mot, alltså har den "vanliga" typen av reset i stort sett spelat ut sin roll.

4. På en 64:a skriver man helt enkelt SYS. Med en Amigan knappar man in sitt program i SEKA och assemblerar en objektfil som sedan är ett fullt körbart maskinkodsprogram bara man skriver dess namn från CLI.

Mata in maskinkod

Kära COMpost,
1. Hur skall jag göra för att mata in "maskinkod"? Man kan ju inte bara sätta på sin 128:a i 64:a läge och skriva LDA $0001 osv.
2. Måste man ha någon speciell "sak" då man programmerar maskinkod?
"En ny maskinkodsprogrammerare"

1. För att mata in maskinkod bör man ha en typ av program som gör bara detta. Denna typ av programm kallas assemblers. De tolkar ditt program och gör om det till ren maskinkod, dvs till bara en massa binära tal i minnet.

Det är även möjligt att använda en maskinkodsmonitor. Med en maskinkods monitor kan man läsa minnet och få det presenterat som mnemonics, dvs instruktionerna typ LDA och STA som är lite lättare att förstå än en radda hexadecimala tal. Med en monitor kan man även editera i ett program. Att skriva längre program med en monitor bör man dock akta sig noga för. Det är lika lätt som att skriva ett BASIC program med radnummermellanrum på 1. Vad händer då du vill lägga in något mellan rad 16 och 17? Svar: Det blir jobbigt!

2. "Saken" du behöver är ett av ovan nämda hjälpmedel, t.ex: Turbo Assembler, Machine Lightning, Macrofire, Laser Genious, Mikro Assembler, J C Assembler, EDNA eller någon annan lämplig assembler. De är dock svåra att få tag på så prova även i användar föreningar.

Monitorer finns det gott om. Alla cartridges har en, och bäst av dessa är Expertens tätt följd av den i Action Replay 5 Professional. 128:an som du äger har dessutom en inbyggd monitor i sitt 128:a läge.

Scrollning av text

Hej! Jag har ett problem med ett program jag gör till 64:an. Hur får man text att flytta sig upp, ner, vänster och höger?
Hälsningar C/S Fans

Man använder sig av en teknik som kallas scrollning. Med två "hårdvaruregister" kan man bestämma den så kallade finscrolls positionen. Det innebär att man kan justera bilden med en precision av en pixel i höjd och sidled. I höjdled används $D011 och i sidled $D016.

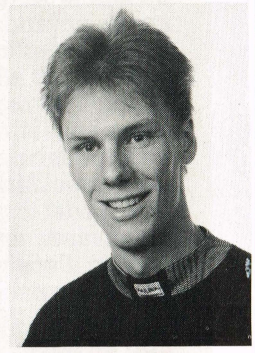
Problemet är att denna finscroll är begränsad till 8 pixels. För att få en mjuk rörelse måste man först låta ett av registerna räkna till max som ju är 7 då man börjar räkna från 0. Vill man nu flytta hela skärmen ett pixelsteg till kan man inte använda dessa mjukscrollregister utan man tvingas istället flytta hela den klump som skall scrollas ett helt teckensteg åt det håll man vill ha den och så återställa scrollregistret till 0.

Detta kallas att grovscrolla på grund av att rörelsens minsta förflyttningssteg är en bokstavs bred/höjd, dvs 8 pixels. Vad man gör är att flytta de enskilda bokstäverna till en ny position. Problemet blir nu att det bara tar ett par cykler att öka scrollregistret medan det tar väldigt lång tid, i maskinkodstid sett, att flytta all data på skärmen vilket gör spelprogrammeringen lite klurigare.

Det är även lämpligt att krympa skärmen både i höjd och sidled för att det inte skall synas att du håller på att mjukscrolla med hela skärmen. Registerna för detta är även denna gång $D011 och $D016.

COMpost

Frågetecknen står i kö, skallen är kliad blodig och lika obegripligt ter sig problemet. Att skrika hjälp rakt ut i luften hjälpte inte heller och ingen enda kan hjälpa i din nöd... Jo vänta, en finns det och det är killen på bilden som utan kostnad tar över kliandet och samtidigt hjälper alla andra med samma problem.



I fortsättningen kommer spalten även att vara öppen för åsikter och vädrade känslor, så skriv in och skapa en storm av dem!

COM post

D Paint III funkar inte på nya A2000

Howdy hackers (som ni brukar säga)! Jag har problem med DeluxPaint III. Jag använde tidigare en A500 utan extra minne vilket ledde till att jag inte kunde använda animeringsbiten. Jag lånade ett extra minne av en kompis och nu gick allt bra. Jag blev faktiskt så biten att jag bestämde mig för att lägga mina surt förvärvade pengar på en sprillans ny Amiga 2000.

Jag blev därför minst sagt förgrymmad då det visade sig att jag inte kunde använda animeringen på min nya maskin. Det såg ut som om jag körde på en oexpanderad 500 modell. Är det något i 2000 maskinen som tar upp en massa minne eller vad är det som är fel.
"Picasso den yngre" från Ludvika

DeluxePaints animerinsdel är knuten till hur mycket fastmem som finns i maskinen. De nya A2000 som släpps nu är utrustade med det nya chippen som klarar av att adressera 1 MB grafik minne och lika mycket chipmem, ett faktum som är utmärkt utom i just detta fall.

DPaint tittar efter och finner att det inte finns något fastmem. Det finns förvisso ett megabyte i maskinen men inte en byte av den typ som Dpaint leter efter minst 512 KB av. Följden är att programet tror sig arbeta i en oexpanderad A500 utan tillräckligt minne att animera i. Misströsta dock inte alla grafiska talanger i landet. Den svenska leverantören HK Elektronics är medvetna om problemet och har larmat Dan Silva.

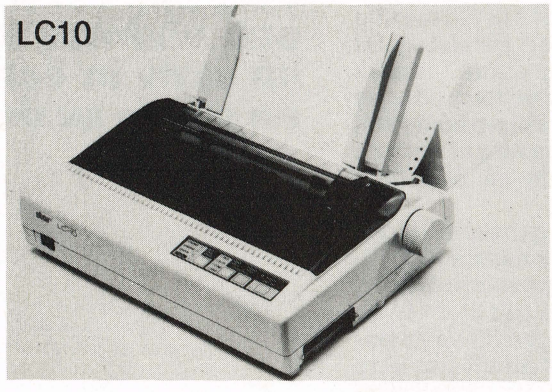
Det kommer en ny version inom kort som saknar denna tragiska defekt.

V irusbeep

Ärade Oberoende COMputer! Går det att ansluta Virusbeepern i extradrivens port för vidare anslutning och i så fall, känner den av både om DF0: och DF1: infekteras?
Ulf Edoff, Linköping

Svaret på båda frågorna är JA! Virusbeepern reagerar på att det skrivs på spår 0 sida 1, oavsett vilken av drivarna detta sker på.

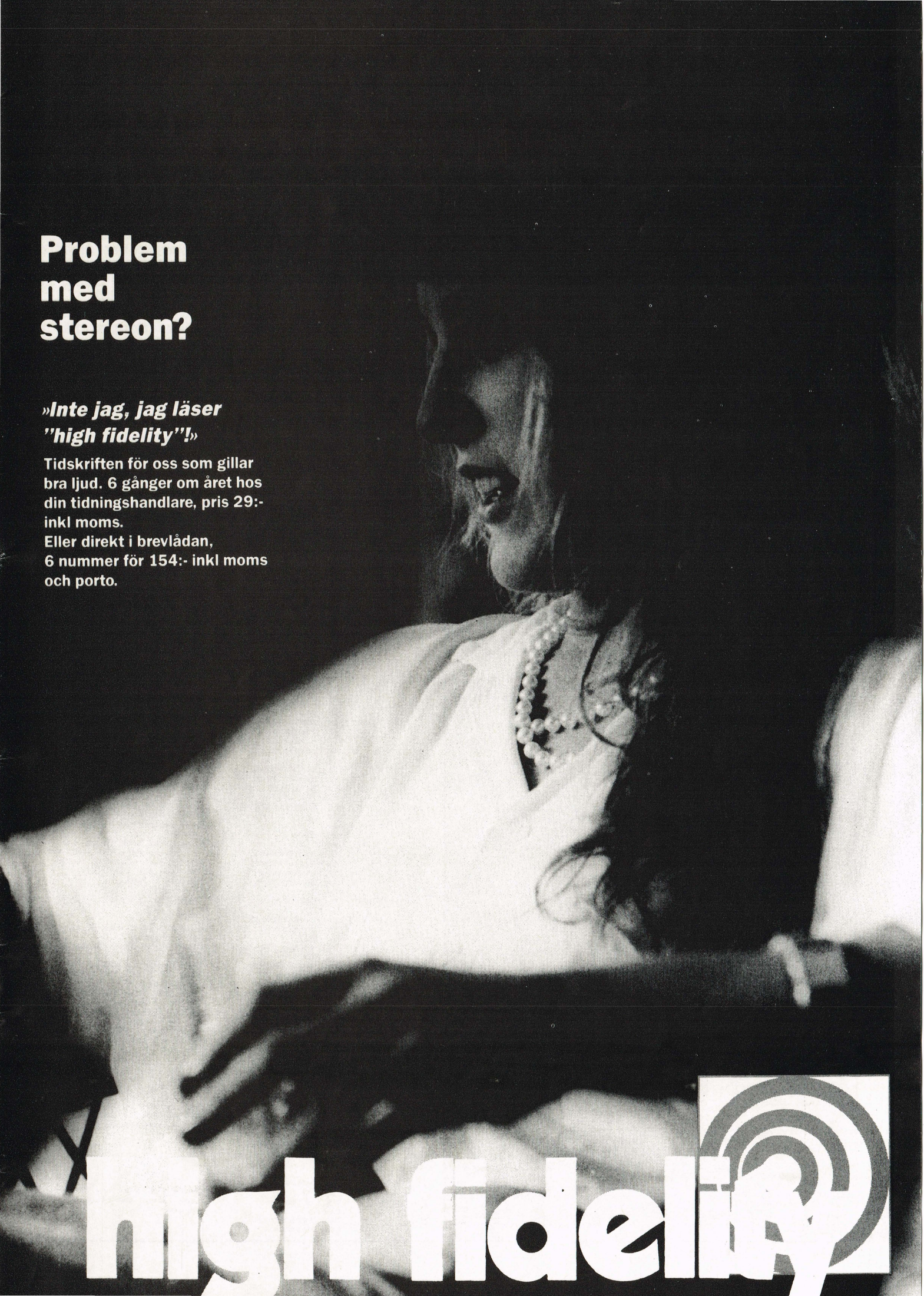
Obeskrivliga skrivare från star til rätt pris hos ADEPT Software



Skrivare från star till din C64, AMIGA, Atari, PC.
Ring för broschyrer, utskriftsprov och prislista

ADEPT Software
Brog. 44
302 38 Halmstad

TEL 035/119668


Problem
med
stereon?
»Inte jag, jag läser
"high fidelity"!»
Tidskriften för oss som gillar
bra ljud. 6 gånger om året hos
din tidningshandlare, pris 29:-
inkl moms.
Eller direkt i brevlådan,
6 nummer för 154:- inkl moms
och porto.
high fidelity

Få disketter med FORMAT

Vad händer egentligen när du formatterar en 1541-diskett? Detta och mycket mer avslöjas i denna artikel, där du ssamtidigt får möjlighet att ändra på allt!

Av Henrik Lund

Vad händer egentligen när du skriver in kommandot OPEN 1, 8, 15, "N:COMputer, 89" och trycker på RETURN?

Det intressanta är vad som händer med kommandot i floppyn, och speciellt dagens tema: Hur utförs formatteringen?

Det hela startar i mainloopen (eller huvudslingan), men det HAR vi skrivit om förr! Jag har "snott" de relevanta maskinkodssnuttarna (se fig 1) direkt från DOSen i min 1541.

Om du ser på $C165 ff, så försöker floppyn här att identifiera det kommando som du skickat. Lyckas den, hoppar den vidare till kommandor (jmp ($6F)). Kikar du på jumptabellen i figur 2, kan du se att det betyder att floppyn hoppar till $EE0D om ett NEW-kommando sändes.

SOURCE 1 :

```
;The following sourcecode is a fastformatter for a C64/C128 and a
;1541/1570/1571.
;
;(C) by Henrik Lund 1987-1989
;This listing is Public Domain
;
;
.eq synczone        =    $05          ;Size of each synczone
.eq spaceheader     =    $0f          ;Delimiterbyte (used in
                                      ; the blockheader)
.eq spacecode       =    $55          ;Bytecode for spacezone
                                      ; on disc = %01010101
.eq synccode        =    $ff          ;Bytecode for SYNC
                                      ;
                                      ;
.eq jobreg          =    $02          ;Jobaddress of our buffer
.eq jobtrack        =    $0a          ;Trackaddress of the buffer
.eq id1             =    $12          ;First byte of the ID
.eq id2             =    $13          ;Second byte of the ID
.eq sectorcount     =    $1b          ;Counter used to count the no.
                                      ; of completed sectors
.eq errorcounter    =    $1f          ;Counteraddress for error during
                                      ; verify
.eq headersign      =    $39          ;Signbyte for the blockheader
.eq checksum        =    $3a          ;Checksum for the datablock
.eq maxsectors      =    $43          ;Max. no. of sectors per. track
                                      ;
.eq blockheaders    =    $0300        ;Buffer, will contains a copy of
                                      ; all blockheaders on a track
.eq headercode      =    $0300        ;The following words indicates
.eq headerchecksum  =    $0301        ; relative location of each byte
.eq headersector    =    $0302        ; in the blockheader (i.e. their
.eq headertrack     =    $0303        ; offset)
.eq headerid2       =    $0304        ;
.eq headerid1       =    $0305        ;
.eq headerfill1     =    $0306        ;
.eq headerfill2     =    $0307        ;
                                      ;
.eq datablock       =    $0700        ;Contains a copy of the data-
                                      ; block
                                      ;
.eq dcporta         =    $1c01        ;PortRegister b for the DC
.eq dcddra          =    $1c03        ;DataDirectionRegister b for the
                                      ; DC
.eq dcpcr           =    $1c0c        ;PortControlRegister
                                      ;
                                      ;
.eq ledon           =    $c100        ;Routine to turn the LED on
.eq closeall        =    $d307        ;Close all channels execept 15
.eq errorafterjob   =    $e60a        ;Errorhandling (errorcode in .A)
.eq calcspeed       =    $f24b        ;Find the no. of sectors on a
                                      ; track and the speed for this
                                      ; track
.eq waitsync        =    $f556        ;Wait for the next SYNC
.eq datatogcr       =    $f78f        ;Convert datablock to GCR
.eq exit            =    $f969        ;Terminate job, errorcode in .A
.eq exitok          =    $fd9e        ;Stop job with an OK
.eq moveup69        =    $fde5        ;Move data in buffer
.eq insertinbuffer  =    $fdf5        ;Insert $01BB-$01FF in current
                                      ; buffer
.eq selread         =    $fe00        ;Change to readmode
.eq headerstogcr    =    $fe30        ;Convert all blockheadere to GCR
                                      ;
                                      ;
                                      ;
                .ba $0500             ;Startadresse of our program in
                                      ; the buffer of the 1541
                                      ;(To prevent malfunction please
                                      ;  do not change this address !)
                                      ;
start           lda  jobtrack         ;Get current track to format
                jsr  calcspeed        ;Get the numbers of sectors
                sta  maxsectors       ;... and save'em
                lda  dcpcr            ;Let the DC change to writemode
                and  #$1f             ;
                ora  #$c0             ;
                sta  dcpcr            ;
                ldx  #255             ;
                stx  dcddra           ;
                lda  #spacecode       ;... and start to erase the disc
                sta  dcporta          ; with "spaces"
                inx                   ;Let sectorcount be zero
                stx  sectorcount      ;
                txa                   ;Make sure that .A and .Y also
                tay                   ; are zero
                                      ;Build new blockheaders
makeheader      lda  headersign       ;Get block-sign
                sta  headercode, y    ;Write it to the buffer
                lda  sectorcount      ;Get sectornumber
                sta  headersector, y  ;Write it to the buffer
                lda  jobtrack         ;Get tracknumber
                sta  headertrack, y   ;Write it to the buffer
                lda  id2              ;Get second byte of ID
                sta  headerid2, y     ;Write it to the buffer
                lda  id1              ;Get first byte of ID
                sta  headerid1, y     ;Write it to the buffer
                lda  #spaceheader     ;Get spacecode for header
                sta  headerfill1, y   ;Write it twice to the buffer
                sta  headerfill2, y   ;
                txa                   ;.X is still zero
                eor  headersector, y  ;Calculate checksum for the
                eor  headertrack, y   ; blockheader by making an
                eor  headerid2, y     ; EOR with ID, track and sector
                eor  headerid1, y     ;
                sta  headerchecksum, y ;Write it to the buffer
                tya                   ;Move index to .A
                clc                   ;Each header has a size of 8
                adc  #8               ; bytes
                tay                   ;Move index back to .Y
                inc  sectorcount      ;... and prepare for next sector
                lda  sectorcount      ;Are through with all block-
                cmp  maxsectors       ; headere for the track ?
                bne  makeheader       ;Not yet : Jump
                txa                   ;.X is still 0 !!
clrdatablock    sta  datablock, x     ;Clear the datablock with 0
                inx                   ;
                bne  clrdatablock     ;
                stx  checksum         ;Checksum is 0 (smart, eh ??)
                stx  $32              ;Set a pointer to 0
                lda  #>(blockheaders) ;Get HI-byte of blockheaders
                sta  $31              ;Save it ...
                jsr  headerstogcr     ;Blockheaders -> GCR
                ldy  #140             ;Move the first 140 bytes in
                jsr  moveup69         ; buffer up
                jsr  insertinbuffer   ;Move data from $01BB-
                                      ; $01FF to buffer
                lda  #>(datablock)    ;Get Hi-byte of datablock
                sta  $31              ;... and save it
                jsr  datatogcr        ;Datablock -> GCR
nextheader      lda  #synccode        ;Write a 5 byte long SYNC
                sta  dcporta          ; to disc
                ldx  #synczone        ;
wrthsync        bvc  wrthsync         ;
                clv                   ;
                dex                   ;
                bne  wrthsync         ;
                ldx  #10              ;Each header is 10 GCR-bytes
                ldy  $32              ;Get pointer to buffer
wrtheader       bvc  wrtheader        ;Write blockheader to
                clv                   ; disc
                lda  blockheaders, y  ;
                sta  dcporta          ;
                iny                   ;
                dex                   ;
                bne  wrtheader        ;
                ldx  #9               ;Write a 9 byte long SPACE-
wrthspace       bvc  wrtspace         ; zone after the block-
                clv                   ; header
                lda  #spacecode       ;
                sta  dcporta          ;
```

Och här börjar den långa resan för vår formatteringsrutin. Figur 3 visar vad som händer från $EE0D och framåt.

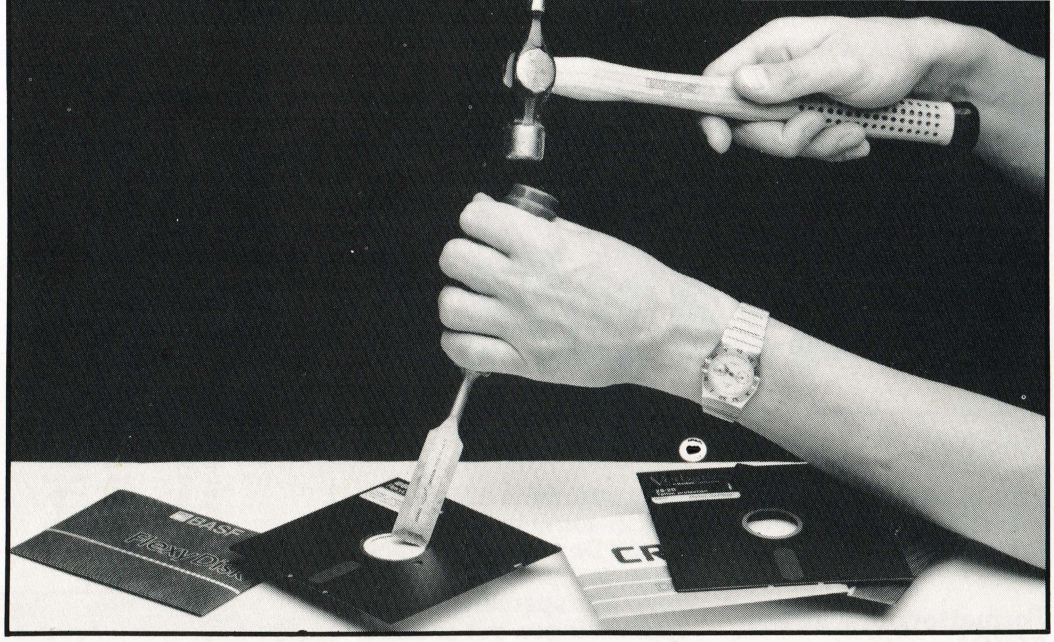
Välj mellan hård och mjuk

Heller inte denna del hör till de mest intressanta. Och ändå... För som du kan se i adress $EE24 ff, är det här som det slutliga valet mellan den "hårda" (långa) och den "mjuka" (korta) formatteringen sker. Och som du kan se drar den "hårda" formatteringsmetoden igenom samma drapa som den "mjuka". Nej, det intressanta är hoppet till $C8C6.

Denna lilla rutin som ligger i $C6C8 "lurar" floppyn. Ta en titt på figur 4. Formatteringsrutinen skriver ned ett litet program (JMP $FAC7) i buffert 3 och startar det sedan som ett JOB ($E0)!

Grunden är ju ganska solklar: Det finns ingen jobbkod för formattering av ett track! Ergo måste floppyn starta sin egen speciella formatteringsrutin på detta lite bakvända sätt. Det här lilla hoppet till $C8C6 och den därav följande starten av jobbet kan vi utnyttja till några rätt speciella effekter.

Eftersom det inte händer någor livsviktigt för floppyn innan jobbet startas (av rutinen i $C8C6), kan vi gri-



```
                dex                     ;
                bne   wrtspace          ;
                lda   #synccode         ;Write a 5 byte long SYNC
                ldx   #synczone         ; to disc
wrtdsync        bvc   wrtdsync          ;
                clv                     ;
                sta   dcporta           ;
                dex                     ;
                bne   wrtdsync          ;
                ldx   #$bb              ;Write the first part of
wrtfirstdata    bvc   wrtfirstdata      ; the datablock to disc.
                clv                     ; This part is located in
                lda   $100, y           ; the area of $01BB-$01FF
                sta   dcporta           ;
                inx                     ;
                bne   wrtfirstdata      ;
                ldy   #0                ;
wrtsecdata      bvc   wrtsecdata        ;
                clv                     ;
                lda   ($30), y          ;Write the second part of
                sta   dcporta           ; the datablock to disc.
                iny                     ; This part is located in
                bne   wrtsecdata        ; the current buffer
                lda   #spacecode        ;Write a 8 byte long SPACE-
                ldx   #8                ; zone after the data-
wrtdspace       bvc   wrtdspace         ; block
                clv                     ;
                sta   dcporta           ;
                dex                     ;
                bne   wrtdspace         ;
                lda   $32               ;Add 10 to the pointer to
                clc                     ; get the start of the next
                adc   #10               ; blockheader
                sta   $32               ;
                dec   sectorcount       ;Are all sectors written down
                                        ; on disc ?
                bne   nextheader        ;Nope : Better jump
waitbyte1       bvc   waitbyte1         ;Wait until the hardware has
                clv                     ; written two more bytes
waitbyte2       bvc   waitbyte2         ; down on the disc
                clv                     ;
verify          jsr   selread           ;Change to readmode
                                        ;If you want to skip the verify-
                                        ; section, please use a JMP END
                                        ; instead of JSR SELREAD
                lda   #200              ;Max. 200 errors are allowed
                sta   errorcounter      ;Save this number
verifytrack     lda   #>(blockheader)   ;Get HI-byte of block-
                                        ; headers
                sta   $31               ;.. and sae it
                lda   maxsectors        ;We are only supposed to test
                sta   sectorcount       ; the number of sectors stored
                                        ; in maxsectors
verifynext      jsr   waitsync          ;Wait for SYNC-signal from
                                        ; blockheader
                ldx   #10               ;Compare the first 10 bytes on
                ldy   #0                ; disc with the contens of the
verifyheader    bvc   verifyheader      ; buffer
                clv                     ;
                lda   dcporta           ;Read a byte from disc
                cmp   ($30), y          ;
                bne   verifyerror       ;Equal : No -> error
                iny                     ;
                dex                     ;Are we through this header ?
                bne   verifyheader      ;No : Read next byte
                clc                     ;Move the bufferpointer 10 bytes
                lda   $30               ; up so that it will point on
                adc   #10               ; next blockheader
                sta   $30               ;... and save it (just in case)
                jmp   verifydata        ;Now let's verify the datablock
verifyerror     dec   errorcounter      ;Decrement errorcounter
                bne   verifysector      ;Is counter > 0: Try again
                lda   #6                ;Terminate job with an error no.
                jmp   exit              ; 6 (24, READ ERROR)
verifydata      jsr   waitsync          ;Wait for SYNC from datablock
                ldy   #$bb              ;Verify data from disc with
                                        ; $01BB-$01FF
verifyfdata     bvc   verifyfdata       ;Wait for next  byte
                clv                     ;
                lda   dcporta           ;... and read it
                cmp   $100, y           ;Compare with original
                bne   verifyerror       ;Equal : No -> report error
                iny                     ;Is part one compared ?
                bne   verifyfdata       ;No : Jump
                ldx   #$fc              ; Compare next part
verifysdata     bvc   verifysdata       ;Wait for byte
                clv                     ;
                lda   dcporta           ;... and read it
                cmp   datablock, y      ;Compare with original
                bne   verifyerror       ;Equal : No -> report error
                iny                     ;
                dex                     ;Are all data checked ?
                bne   verifysdata       ;No : Jump
                dec   sectorcount       ;Are all sectors checked ?
                bne   verifynext        ;No : Verify the next sector
end             jmp   exitok            ;Terminate job with an OK
                                        ;
                                        ;
fastformat      ldy   #0                ;Move name, ',' og ID into the
copynameid      lda   name, y           ; inputbuffer
                sta   $200, y           ;
                iny                     ;
                cpy   lengthofname      ;Was this the last part of the
                                        ; name ?
                bcc   copynameid        ;No : Jump
                lda   lengthofname      ;Save the total length of name
                sta   $274              ; + ',' og ID
                lda   startofid         ;Get start of ID
                sta   $27b              ;... and save it
                lda   #0                ;Let default drive be
                sta   $7f               ; 0 (standard)
                jsr   ledon             ;Turn LED on
                ldy   $27b              ;Move the ID into it's normal
                lda   $200, y           ; location
                sta   id1               ;
                lda   $201, y           ;
                sta   id2               ;
                jsr   closeall          ;Close all channels (except the
                                        ; commandchannel)
                lda   #$14              ;Increase the number of inter-
                sta   $1c07             ; rupts per second
                lda   #$c0              ;Perform a BUMP
                sta   jobreg            ;
waitjobbump     lda   jobreg            ;Wait on termination of BUMP-job
                bmi   waitjobbump       ;
                ldx   firsttrack        ;Get first track
nexttrack       stx   jobtrack          ;Save track in trackaddress
                lda   #$e0              ;Perform our job for this track
                sta   jobreg            ;
waitjobdoit     lda   jobreg            ;Wait for termination of job
                bmi   waitjobdoit       ;
                cmp   #2                ;Any errors ?
                bcs   errorfromjob      ;If so STOP
                inx                     ;Increment current track by 1
                cpx   lasttrack         ;Has every track been formatted?
                bcc   nexttrack         ;No : Here we go ...
                jmp   $ee40             ;Terminate formatter by writing
                                        ; a new directory and BAM to
                                        ; disc
errorfromjob    tax                     ;Compute error as
                jmp   errorafterjob     ;      20 + .X - 2
                                        ;ex. 6 => 20 + 6 -2 =
                                        ;         = 24, READ ERROR
                                        ;
firsttrack      .by   $00               ;First track to format
lasttrack       .by   $00               ;Last track
                                        ;
lengthofname    .by   $00               ;Total length of name, ',' and
                                        ; ID
                                        ;
startofid       .by   $00               ;Place where to find the ID
                                        ;
name            .tx   STRING(.19.)      ;Name and ID of disc
                                        ; Please remember to seperate
                                        ; by a ','
```

Få disketter med FORMAT

pa in och själva kalla på formatteringsrutinen. Det väntar vi dock lite med.

Formatteringsrutinen $FAC7

Vad händer i själva formatteringsrutinen i $FAC7?

Rutinen är uppdelad i flera delar:

1) Initialisering $FAC7-$FAFF. Först kollas det i adressen $51. Är denna adress större än 127 (normalt står det $FF i den), betyder det att formatteringen precis påbörjats programmet sörjer för att initialisera olika adresser och utför ett BUMP av läs/skrivhuvudet (det karaktäristiska ra-ta-ta-ljudet du hör vid varje formattering).

Om formattering redan är igång ($51 är mindre än 128) kollar floppyn om det nuvarande spåret är lika med det sist formatterade spåret. Är de lika hoppar floppyn vidare till $F99C för att positionera läs/skrivhuvudet. Är de inte lika fortsättes det med del 2:

2) Uppmätning $FB00-$FC35. Detta är den absolut mest tidskrävande delen av hela formatteringen. Disketten delas först upp i två zoner, en SPACE-zon ($55) och en SYNC-zon ($FF). Sedan mäts spåret noggrant så att floppyn kan få veta hur många byte som kan "stå" på spåret. Men varför nu detta?

Jo, mätningen ska se till att sektorerna blir regelmässigt fördelade över spåret. Jag berättade en gång att en sektor har ett helt bestämt utseende: En SYNC, därefter en blockheader, en SPACE-zon, en SYNC och sedan själva datablocket. Alla dessa har storlekar som är lika för varje sektor på hela disketten. Det enda som fattas nu är en smula "lebensraum" till nästa sektor.

För att få detta tillfogas en SPACE-zon innan nästa sektor. Denna zon har dock ingen fast storlek utan varierar från spår till spår. Det uppmätningen gör är alltså att ta reda på hur stort detta "hål" mellan de enskilda sektorerna ska vara.

3) RAM-formatteringen $FC36-$FCAD. Det nästa floppyn gör är att lägga en "arbetskopia" av samtliga blockheaders på det aktuella spåret i buffert 0 ($0300- $03FF) så som de så småningom ska se ut på disketten.

Dvs. varje blockheader byggs upp från grunden med blockheaderkännetecknet, checksum, track- och sektornummer, ID samt de två loopbyte ($0F). Dessa konverteras så till GCR varefter floppyn börjar att bygga up datablocket.

4) Beskrivning $FCAE-$FD23. Nu blir spåret allra först raderat med rutinen i $FE0E, varefter varenda sektor skrivs ut på disketten. Detta sker naturligtvis på "normalt" sätt via STA $1C01, och i den ordningsföljd som sektorn är uppbygg dpå (alltså först SYNC, sedan blockheader osv.).

5) Verifikation $FD24-$FD8A. Det första som händer är att floppyn skiftar till READ-mode via rutinen i $FE00. Sedan kontrolleras det att varje sektor har skrivits korrekt ned på disketten.

Det kan maximalt uppstå 200 läsfel (direkta fel på disketten eller försök att läsa fel header) innan floppyn stoppar formatteringen och ger felmeddelande.

6) Avslutning $FD8B-$FDA2. Innehållet av adressen $51 räknas upp med 1 och jämförs med värdet 36. Är det mindre än 36 hoppas det vidare till $F99C, varvid det sker en ny start av formatteringen i $FAC7. Om

FIGUR 1 :

```
..ebe7  58        cli             ;Start of mainloop
..ebe8  ad 00 18  lda  $1800      ;Put ATN ACK, CLOCK and DATA high to
..ebeb  29 e5     and  #$e5       ; define busstate
..ebed  8d 00 18  sta  $1800      ;
..ebf0  ad 55 02  lda  $0255      ;Any command present ?
..ebf3  f0 0a     beq  $ebff      ;No : Continue looping
..ebf5  a9 00     lda  #$00       ;Yes : Delete commandflag
..ebf7  8d 55 02  sta  $0255      ;
..ebfa  85 67     sta  $67        ;Delete NMI-condition flag
..ebfc  20 46 c1  jsr  $c146      ;Check and execute command
..ebff  58        cli             ;
..ec00  a5 7c     lda  $7c        ;Check for ATN from C64
..ec02  .. .. ..  ...  .....     ;.....................


..c146  a9 00     lda  #$00       ;Value "BAM-not-dirty"
..c148  8d f9 02  sta  $02f9      ;Make BAM "clean"
..c14b  ad 8e 02  lda  $028e      ;Get standarddevice (0)
..c14e  85 7f     sta  $7f        ;... and save as current device
..c150  20 bc e6  jsr  $e6bc      ;Prepare for OK
..c153  a5 84     lda  $84        ;Get secondaryaddress
..c155  10 09     bpl  $c160      ;Secondaryaddresse < 128 ??
..c157  29 0f     and  #$0f       ;Look only at channels 0-15
..c159  c9 0f     cmp  #$0f       ;OPEN file, dev, 15 ??
```

SOURCE 2 :

This routine can be used to format single tracks in the area of track 1-41. Track in the area 35-41 are been formatted with speed 0 and 17 sectors

```
jobtask     jmp   doformat        ;Jump to rutine (will be modified
                                  ; by the program
doformat    ldx   track           ;Get track to be formatted
            cpx   #36             ;Is it beyond 35 ??
            bcc   normal          ;No : Jump
            lda   $1c00           ;Set speed for tracks larger than
            and   #$9f            ;track 35. Speed is 0
            sta   $1c00           ;
            lda   #17             ;Set no. of sectorer to 17
            sta   $43             ;
normal      stx   $51             ;Save the trackumber
            lda   #$0a            ;The following 6 lines are only
            sta   $0620           ; included to make the routine
            lda   #$a0            ; 100% legal. You may exclude
            sta   $0621           ; them if you want to
            lda   #$0f            ;
            sta   $0622           ;
            lda   #<(format)      ;Set JMP to start of format-
            sta   jobtask + 1     ; routine
            jmp   $fb00           ;Start formatting
format      lda   #<(doformat)    ;Restore JMP to it's startvalue
            sta   jobtask + 1     ;
            jmp   $fd9e           ;End job with an OK
                                  ;
            ldy   #firsttrack     ;Get first track to format
nexttrack   sty   track           ;Write it into jobadress
            lda   #$e0            ;Get code for "STARTJOB"
            sta   jobadr          ;Start jobbet
wait        lda   jobadr          ;Get reply from job
            bmi   wait            ;Wait until job is through
            iny                   ;Increment current track
            cpy   #lasttrack + 1  ;Has every track been formatted
            bne   nexttrack       ;No : Jump
            rts                   ;Finito, back to where we came from
```

SOURCE 3 :

This routine can be used to write a new. Remember that NEWNAME consists of the new discname PLUS a comma (',') as a delimiter

```
            ldy   #$10            ;16 characters plus a comma gives $10
            sty   $0274           ;Save length of name
loop        lda   newname, y      ;Move name into inputbuffer
            sta   $0200, y        ;
            dey                   ;Repeat these steps until all characters
            bpl   loop            ; have been moved
            jmp   $ee40           ;Write a new directory
```

SOURCE 4 :

The following routine replaces the B-R (U1) command. This routine is able to read tracks above track 35.
B-W (U2) can be implementated by replacing JMP $F4D1 with JMP $F575

```
                lda  track        ;Get current track
                cmp  #36          ;Is it above track 35 ?
                bcc  normal       ;No : Jump
                lda  $1c00        ;Set speed for track 36-41 to 0
                and  #$9f         ;
                sta  $1c00        ;
normal          lda  #>(buffer)   ;Get HI-byte of buffer
                sta  $31          ;Save it for the routine
                jmp  $f4d1        ;Read sector into the buffer

                lda  #ntrack      ;Track to be read
                sta  track        ;Save it as jobtrack
                lda  #nsector     ;Sector to be read
                sta  sector       ;Save it too ...
                lda  #$e0         ;Get code for "STARTJOB"
                sta  jobadr       ;Start job
wait            lda  jobadr       ;Get reply from job
                bmi  wait         ;Wait until job is through
                rts               ;Finito, back to where we came from
```

```
..c15b  f0 03     beq  $c160      ;YES SIR : Find command ...
..c15d  4c b4 d7  jmp  $d7b4      ;No : Perform a normal OPEN file
..c160  20 b3 c2  jsr  $c2b3      ;Initialize tables
..c163  b1 a3     lda  ($a3), y   ;Get first character in $0200
..c165  8d 75 02  sta  $0275      ;Save it until later
..c168  a2 0b     ldx  #$0b       ;Put max. no. of commands in .X
..c16a  bd 89 fe  lda  $fe89, x   ;Get command from table
..c16d  cd 75 02  cmp  $0275      ;Compare with saved value
..c170  f0 08     beq  $c17a      ;Found ? Yes : Exit
..c172  ca        dex             ;Decrement .X
..c173  10 f5     bpl  $c16a      ;Was this the last command : No
..c175  a9 31     lda  #$31       ;ERROR : Report a "31, SYNTAX ERROR"
..c177  4c c8 c1  jmp  $c1c8      ;
..c17a  8e 2a 02  stx  $022a      ;Save the number of the command
..c17d  e0 09     cpx  #$09       ;Is the command requiring an
..c17f  90 03     bcc  $c184      ; additional  filename ? No : Jump
..c181  20 ee c1  jsr  $c1ee      ;Parse command for '=', ':' and ','
                                  ; to identify filenames etc.
..c184  ae 2a 02  ldx  $022a      ;Get commandnumber
..c187  bd 95 fe  lda  $fe95, x   ;Get LO-address of command and save
..c18a  85 6f     sta  $6f        ; it in $6f
..c18c  bd a1 fe  lda  $fea1, x   ;Get HI-adresse of command and save
..c18f  85 70     sta  $70        ; it in $70
..c191  6c 6f 00  jmp  ($006f)    ;Perform an indirect jump to the
                                  ; program which $6f/$70 is pointing
                                  ; at
```

FIGUR 2 :

```
Commandtable
..fe89  56        V   VALIDATE    ;This is the commandtable
..fe8a  49        I   INITIALIZE  ;
..fe8b  44        D   DUPLICATE   ;(Is NOT used !!)
..fe8c  4d        M   MEMORY      ;
..fe8d  42        B   BLOCK       ;
..fe89  55        U   USER        ;
..fe89  50        P   POSITION    ;
..fe89  26        &   &-kommando  ;
..fe89  43        C   COPY        ;The following commands requi:
..fe89  52        R   RENAME      ; additional "filename"
..fe89  53        S   SCRATCH     ;
..fe89  4e        N   NEW         ;


Jumptable :
..fe95 84         fea1 ed         ;V  VALIDATE
..fe96 05         fea2 d0         ;I  INITIALIZE
..fe97 c1         fea3 c8         ;D  DUPLICATE  ( - )
..fe98 f8         fea4 ca         ;M  MEMORY
..fe99 1b         fea5 cc         ;B  BLOCK
..fe9a 5c         fea6 cb         ;U  USER
..fe9b 07         fea7 e2         ;P  POSITION
..fe9c a3         fea8 e7         ;&  &-Command
..fe9d f0         fea9 c8         ;C  COPY
..fe9e 88         feaa ca         ;R  RENAME
..fe9f 23         feab c8         ;S  SCRATCH
..fea0 0d         feac ee         ;N  NEW
```

FIGUR 3 :

```
..ee0d  20 12 c3  jsr  $c312      ;Set driveparameters
..ee10  a5 e2     lda  $e2        ;Get drivenumber
..ee12  10 05     bpl  $ee19      ;Continue if drive OK
..ee14  a9 33     lda  #$33       ;Otherwise report "33, SYNTAX ERROR"
..ee16  4c c8 c1  jmp  $c1c8      ;
..ee19  29 01     and  #$01       ;Isolate drivenumber
..ee1b  85 7f     sta  $7f        ;Let it be the new standard
..ee1d  20 00 c1  jsr  $c100      ;LED on
..ee20  a5 7f     lda  $7f        ;Get drivenumber
..ee22  0a        asl             ;Multiply by two to get index
..ee23  aa        tax             ; to the table
..ee24  ac 7b 02  ldy  $027b      ;Get startposition of ID
..ee27  cc 74 02  cpy  $0274      ;Compare it with the total length of
                                  ; the command
..ee2a  f0 1a     beq  $ee46      ;If equal, ID is missing : Jump
..ee2c  b9 00 02  lda  $0200, y   ;Move ID to zeropage of floppy so
..ee2f  95 13     sta  $12, x     ; that it can be remembered
..ee31  b9 01 02  lda  $0201, y   ;
..ee34  95 13     sta  $13, x     ;
..ee36  20 07 d3  jsr  $d307      ;Close all channels, except 15
..ee39  a9 01     lda  #$01       ;Set first track to format to
..ee3b  85 80     sta  $80        ; track 1
..ee3d  20 c6 c8  jsr  $c8c6      ;Start formatting
..ee40  20 05 f0  jsr  $f005      ;Erase BAM-buffer
..ee43  4c 56 ee  jmp  $ee56      ;Continue formatting
..ee46  20 42 d0  jsr  $d042      ;Initialize disc
......  .. .. ..  ...  .....     ;
```

FIGUR 4 :

```
..c8c6  a9 4c     lda  #$4c       ;Write 'JMP $FAC7' to buffer 3
..c8c8  8d 00 06  sta  $0600      ; ($0600-$6FFF)
..c8cb  a9 c7     lda  #$c7       ;
..c8cd  8d 01 06  sta  $0601      ;
..c8d0  a9 fa     lda  #$fa       ;
..c8d2  8d 02 06  sta  $0602      ;
..c8d5  a9 03     lda  #$03       ;Set the actual track- and sector-
..c8d7  20 d3 d6  jsr  $d6d3      ; number for buffer 3
..c8da  a5 7f     lda  $7f        ;Get actual drivenumber
..c8dc  09 e0     ora  #$e0       ;EOR with jobcode (to get real job)
..c8de  85 03     sta  $03        ;Start job
..c8e0  a5 03     lda  $03        ;Get reply from job
..c8e2  30 fc     bmi  $c8e0      ;Wait until job is finished
..c8e4  c9 02     cmp  #$02       ;Is it an OK ??
..c8e6  90 07     bcc  $c8ef      ;Yep : Jump
..c8e8  a9 03     lda  #$03       ;Errornumber
..c8ea  a2 00     ldx  #$00       ;Track- and sectornumber kan be
                                  ; found in the jobaddress for buffer
                                  ; 0
..cec8  4c 0a e6  jmp  $e60a      ;Report an 21, READ ERROR
..c8ef  60        rts             ;Ok, terminate format
```

det är lika med 36 stoppas formatteringen med ett OK.

Efter att ett BUMP utförts eller efter att ett track formatterats sker det alltså ett hopp till $F99C, där det åter undersöks om det finns något jobb på gång. Då jobbkoden $E0 (eller $C0) inte raderas förrän formatteringen är klar, betyder det att rutinen i $0600 (och därmed i $FAC7) kallas gång på gång till formatteringen (av en eller annan orsak) är klar.

Puh, det var en tung omgång!

Uppmätning av spår

När vi nu HAR kikat på den gamla formatteringsrutinen, så kunde vi ju passande försöka förbättra den. Som jag tidigare har nämnt, har motorn inne i 1541:an ett varvtal på 300 rpm (varv per minut) eller 5 varv per sekund. Teoretiskt sett borde man alltså kunna formattera ett spår på 0.2 sekunder! Eller en hel diskett på 7 sekunder!!

Varför är då 1541:an hela 10 gånger långsammare? Den största flaskhalsen ligger helt klart i att vartenda spår ska mätas upp noggrant så att alla sektorer placeras med samma avstånd runt om på disketten.

Internt i 1541:an finns en regel om att en SPACE-zon mellan två sektorer INTE får vara mindre än 4 bytes lång. Då jag under tidens lopp har arbetat med en hel del olika 1541-drivar har jag lagt märke till att den minsta SPACE-zonen normalt ligger på hela 9 bytes!

Vi kan alltså bespara oss från den långsamma uppmätningen genom att utgå från ett standardavstånd mellan de enskilda sektorerna på t ex 8 bytes. Vi kunde också se lite närmare på den 110% radering som i starten utförs på varje spår. Denna är faktiskt inte nödvändig.

Om du en dag sätter dig ned med en fickräknare och räknar ut hur många sektorer det kan vara på ett spår, kommer du fram till något som likna 1/2 till 1 sektor EXTRA per spår. Då en GCR-(de)kodning av en enda buffert tar nästan 0.02 sek, eller den tid som det tar för att susa iväg över TO sektorer (0.2 sek per spår och 20 sektorer), så når vi fram till följande:

Börjar vi med att radera INNAN vi påbörjar en GCR-kodning genom att skriva $55 till disketten, kommer vi EFTER kodningen att har raderat så mycket av disketten (ett område motsvarande två sektorer) att vi kan börja att skriva till disketten.

Ta en titt på SOURCE 1. Det är faktiskt (nästan) en ren kopia av den formatteringsrutin som du hittar i din 1541:as DOS. Jag har bara inkluderet ovanstående finesser så att formatteringstiden nu är nere på ca 20 sekunder! Alltså fyra gånger snabbare än originalet!

Ett jobb-program

Så var det ju det där med att $FAC7 var ett jobb-program. Och visst kan vi utnyttja det. SOURCE 2 är ett litet program som kan formattera enskilda spår på disketten. Du kan t ex använda det för att formattera spår 36-41 eller 2-6.

SOURCE 4 kan du så använda för att skriva och läsa enskilda sektorer på dessa spår. Kom bara ihåg att använda I-kommandot innan programmen för att säkra att innehållet i $12 och $13 i floppyn är up-to-date.

Om du "av misstag" raderar spår 18 kan du använda SOURCE 3 för att skriva en ny directory. Kom bara ihåg att du SJÄLV ska skriva ner RÄTT ID i $12 och $13. För övrigt kommer du säkert snabbt att återfinna SOURCE 3 i SOURCE 1. Kom ihåg att låta namnet efterföljas av ett komma "," på båda ställena, annars fungerar det inte så bra.

Den magiska gränsen

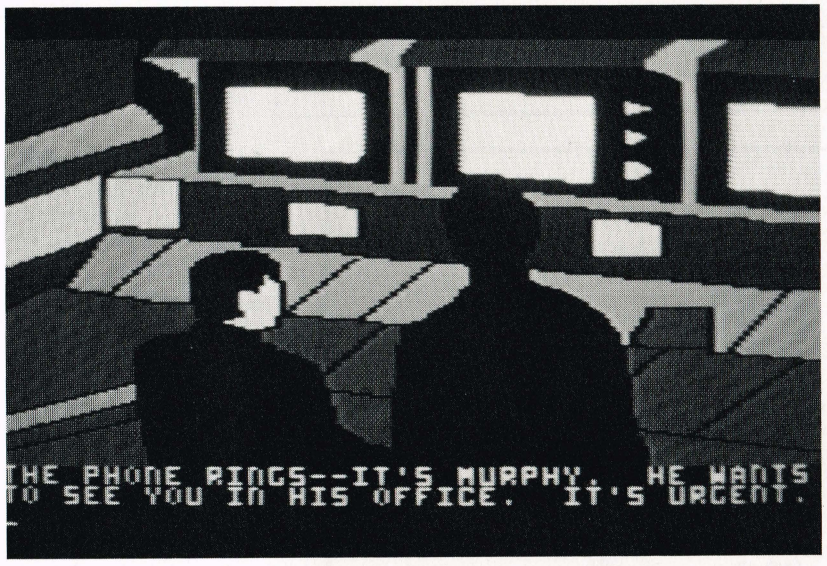
Hur kan man då göra för att formattera en diskett på 8 sekunder?

Bra fråga. Tiden ligger bara 1 sekund över den teoretiskt möjliga. Jag tänker ge dig en chans att fundera över det till nästa gång (prova t ex att ta bort den långsamma biten, jobbslingan).

Skulle du få lust att exprimentera med formatteringsrutiner är det bara att sätta igång. Men använd SOURCE 1 som grund när du provar dina olika idéer. Tills vi ses igen vill jag bara önska riktigt mycket nöje och förhoppningsvis inte alltför många felformatterade disketter.

Lär Dig Att Lösa Äventyr

Så löser du äventyrsspel! The Dungeon Master berättar allt om de hemliga trix som gör att du kan klara dig igenom även de mest motsträviga äventyren. Läs, begrunda och bli den nye mini-TDM!



Äventyrsfirman Telarium baserade sina äventyr på böcker av kända författare, bl a Arthur C Clarkes "Rendezvous with Rama" och Ray Bradburys "Fahrenheit 451". Telarium hade den sällsynta förmågan att kunna blanda ljud, grafik och text till en vidunderlig helhet, och Telariumäventyr hör fortfarande till äventyrsgenrens höjdpunkter - "Amazon" (bilden) är en av dem.

Av - The Dungeon Master!

Du har säkert många gånger undrat över hur äventyrsexperter som The Dungeon Master och andra egentligen bär sig åt - de verkar ju kunna lösa alla äventyr de får syn på utan några större problem. Hur bär de sig åt? Är det något speciellt, har de tumme med högre makter, kort sagt - är de geniala?

Svaret är - ja.

Men TDM har förbarmat sig och kommer i denna och en följande artikel att lyfta på slöjorna för några av de metoder som används i den innersta magiska kretsen när de ska "betvinga" ett svårt äventyr.

Vad är ett äventyr?

Först måste vi ha en sak helt klart för oss: Vad är egentligen ett "riktigt" äventyrsspel? Det finns flera hållningar till denna fråga, och de senaste åren har några trolösa individer dessutom vågat att räkna in rollspel, Role-Playing i äventyrens rad.

Rollspel är egentligen bara en utökad version av Monopol, där ens öde bestäms av tärningskast! Jag gillar att spela Monopol, men något äventyr är det knappast!

Ett "riktigt" äventyr består antingen av enbart text eller av text tillsammans med grafik och/eller ljud. Kommandona skrivs in och joysticks är bannlysta. Så är det bara.

Alla spelar äventyrsspel

Det finns ingen åldersgräns för datorspelande. Ändå är det mest yngre som sitter ock spelar de senaste grafikexplosionerna och läser AMIGAMES eller Soft Check.

Annat är det med äventyr. Här är det nästan lika många vuxna som ungdomar som sitter framför tangentbordet. Det rör sig ju inte om att vara snabbast på joystickknappen, utan mer om att vara klok, uppfinningsrik, ha tålamod och ha ett ordentligt ordförråd på engelska. De flesta äventyren är ju skrivna av amerikaner eller engelsmän, och det var också i USA som de första äventyren skapades och sattes i produktion.

Begynnelsen

Det hela började i USA, där några universitetsstuderande gjorde föregångaren till det vi idag känner som Colossal Cave Adventure på universitetets stora mainframe-dator. Detta var världens första riktiga äventyrsspel.

En av männen bakom det hela, Joel Berez, förde idén vidare och startade den firma som idag är välkänd under namnet Infocom. Firman fungerar något annorlunda nu, men i början producerade den enbart äventyrsspel - och tjänade bra på det. Nästan samtidigt startade Scott Adams, även han i USA, med att göra äventyr, de berömda/beryktade Hulk, Spiderman m fl.

Där Infocom satsade på atmosfär och en spännande historia, var Scott Adams äventyr ett enda stort pussel, mer baserade på direkta gåtor än någon egentlig bakgrundshistoria. Adams har så småningom blivit något av en kultfigur, antingen som hatobjekt eller som äventyrsguru. Jag har löst de flesta av hans äventyr, men något nöje var det knappast!

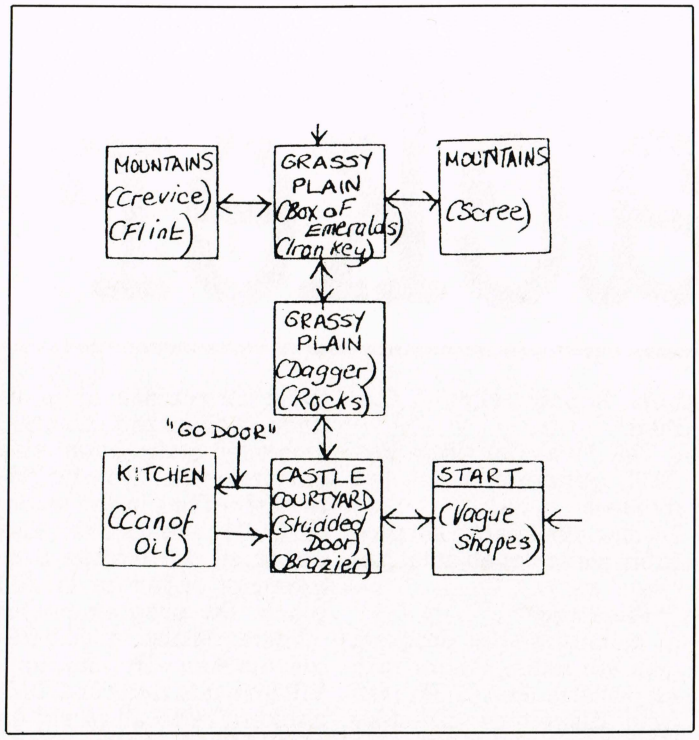
Vad krävs det?

Det finns tre, ja faktiskt fyra viktiga huvudelement som måste finnas framme i samband med äventyrslösning:

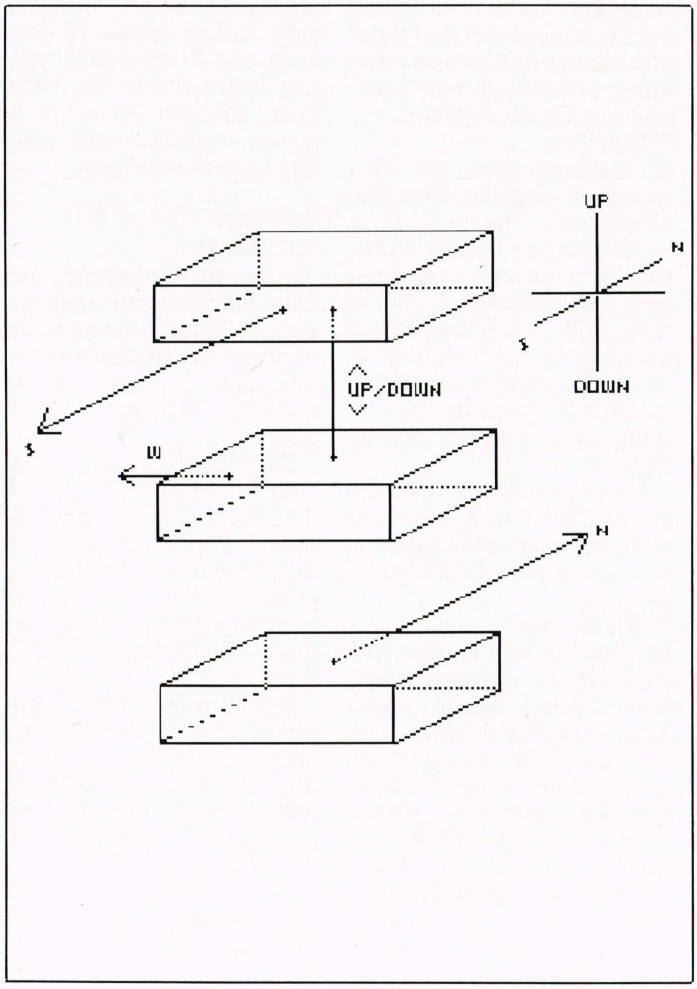
Blyerts, papper, suddgummi och ordbok. När du sitter och ska lösa ett äventyrsspel utgör nämligen kartläggningen 50% av arbetet. Man kan också tillfoga ytterligare ett huvudelement, och det är att ha en god vän vid sin sida. Det kan ha sina fördelar att låta två hjärnor rådbråkas om samma problem samtidigt.

Så ritar du kartor

En karta över ett äventyr kan tillverkas på två sätt: I två dimensioner (den mest använda metoden, se fig 1) samt i tre dimensioner (se fig 2).



Den lättaste kartläggningsmetoden är i två dimensioner. Utgångarna kan vara lite ologiska, men det finns massor med plats till att notera ev ting och saker vid varje rum och annat nyttigt.



Tredimensionell kartritning är den som tar längst tid - men är också både den mest logiska och den mest oöverskådliga. Vissa firmor, bl a Infocom, använder sig av just denna metod när de bygger upp sina äventyr.

Bäst är att göra kartan i tre dimensioner, men det tar i gengäld också betydligt längre tid.

Samtidigt som man kartlägger själva äventyret, är det också en god idé att notera var man funnit olika saker och var man eventuellt lagt ifrån sig dem igen.

Att kartlägga en labyrint är egentligen inte så svårt. Man tar allt man kan skrapa ihop med sig in i labyrinten och lägger så en sak i varje nytt rum man kommer till. Kom ihåg att ett rum kan ha upp till 12 olika in- och utgångar (n, s, e, w, nw, ne, se, sw, d, u, in, out). Normalt är "in" eller "out" också en kompassriktning.

I vissa äventyr kan du dock bara komma till vissa rum med hjälp av spells (t ex "Spellbreaker"), men då är det bara att notera vilken spell det rör sig om vid respektive rum.

Det är alltid bättre att göra en för stor karta än en för liten. Bred alltså ut dig över flera A4-sidor om det behövs.

Ordförråd

Antal ord och vilka ord ett äventyr förstår är helt olika från äventyr till äventyr. Ord som Get, Take och Drop är väl de ord som går mest igen. Tyvärr existerar ingen standard för ordförråd i äventyr. Det finns sorgliga exempel på äventyr som förstår Get men inte Take eller omvänt.

I sådana fall är det helt enkelt nödvändigt att skriva ner de ord som datorn förstår. Du blir alltså tvungen att prova dig fram via olika ordkombinationer. Antingen förstår datorn, eller också får du ett meddelande i stil med "I don't understand the word XXX"!

Idag är det bara de mest primitiva äventyren som inte förstår alla grundorden. Stöter du ändå på ett kan man råda bot på det ändå - man kan fuska! Men det återkommer jag till senare.

SAVE ofta

Göm din position ofta. Det bästa är att SAVE precis före och precis efter ett pussel. Då borde problemet vara täckt in. Har du möjlighet att göra SAVE i RAM - använd dig av den möjligheten, så slipper du sitta och vänta på diskettstationen.

Spelar du t ex ett Infocomäventyr på Amiga (som startas från CLI), gömmer du bara dina savings i RAM. Där blir de kvar tills du stänger av eller resettar maskinen. När du inte vill spela längre, hoppar du bara ut i CLI och kopierar över RAMen på en SAVE-diskett.

Genialt i all sin enkelhet.

En SAVING kan man även gå in och rota i lite efteråt och ändra lite parametrar, om man har en diskmonitor.

Ofta är SAVINGS tämligen oöverskådliga, med massor med kryptiska tecken och hexadecimaler överallt. Men man kan ju alltid be en bön om att man har träffat rätt och ändrat lite rätt saker, så att man nästa gång man läser in sin SAVING har massor av det ena eller eventuellt det andra.

Detta arbetssätt är dock riktigt jobbigt och bör bara användas om man inte har annat för sig - som t ex att koncentrera sig om att lösa äventyret!

Var får man tag på äventyr

Normalt kan man köpa äventyr hos den lokala handlaren - annars kan han skaffa dem, om han giter.

Alternativt kan man skaffa äventyr direkt från firmorna i England, men det kan dröja ett tag innan de kommer fram.

Det finns också en del "underground"-äventyr som inte går att få tag på i handeln. Det finns nämligen massor med äventyrsklubbar i utlandet, där medlemmarna själva tillverkar egna äventyr. Dessa kan vara riktigt bra, men samtidigt också svåra att på tag på för en utomstående.

Nästa gång

Om du sitter riktigt ordentligt fast i ett äventyr, är som sagt en av möjligheterna att använda sig av lite ofina metoder. I nästa artikel kommer jag att avslöja hur man kan fuska i flera äventyr - och med exempel så att du själv kan prova direkt.

Vi ska bl a titta på äventyr från Channel 8, Level 9, Telarium och äventyr skrivna med The Quill, samt hur man kan smygtitta i ett par Amigaäventyr.

Och är fusk inget för dig kan du ju bara låta bli att läsa nästa spännande artikel - för DU fuskar väl inte...?

Knappa in:

COMpacker

Om ni läste förra numret i serien om datapackning vet ni vad en packer är. Nu har Oberoende COMputer nöjet att presentera sin alldeles egen packer som ni kan använda att packa era egna program med. En charpacker i världsklass får man inte till skänks varje dag, så läs och knappa in "COMpacker 1.0". Det är väl värt besväret.

Lite repetition

Vi tar lite repetition till att börja med. En charpacker är ett program som förkortar bort långa rader av likadana bytes till en kort signalbyteskombination. Vår packer arbetar t.ex. efter principen att man gör den långa raden till en trebyteskombination enligt modellen: signalbyte, antal, byte.

Signalbyten är den byte som talar om för uppackaren att den skall tolka de två följande bytesen som antal och byte istället för att bara representera sig själva. En rad med $24 stycken $42 bytes på rad kan t.ex förkortas $C6, $24, $42, om man förutsätter att signalbyten är $C6.

Signalbytens funktion

Vad som utmärker signalbyten är att det bör vara den byte som förekommer minst antal gånger i koden. Varför skall den då förekomma så få gånger som möjligt? Jo, eftersom man måste låta signalbyten följas av några speciella parametrar för att man skall veta att den i detta fall representerar sig själv och inte är en signal byte.

Vår packer låter signalbyten följas av $00, $00 om signalbyten själv skall förekomma i den packade koden. Detta medför en förlust på två bytes. Man är därför angelägen om att denna förlust skall inträffa vid så få tillfällen som möjligt.

Man letar alltså upp den byte som förekommer minst antal gånger och kallar den för signalbyte. I vårt fall visas kalkyleringen av aktuell signalbyte på skärmen och resultatet kommer att blinka när filen är inladdad och färdiganalyserad.

En packers funktion

En packer har sitt huvudsakliga användningsområde då man snabbt skall packa en mängd data. Alla på marknaden existerande freezers har någon form av packer i sig för att man inte skall spara ut riktigt hela minnet direkt på disk.

Det skall vara en något mindre, men dock komplett version av det som fanns i minnet, för att man skall får plats med mer än två program på disken. Packers byggs numera även in i både grafik- och musikrutiner för att musik- och levelgrafikdata skall bli så volymässigt små som man kan få dem. Detta leder givetvis till att mera sofistikerade spel kan trängas in på det begränsade området.

Skillnaden blir att uppackningen i dessa rutiner sker under tiden programmet kör, medan vår version, i likhet med marknadens andra charpackers, packar ihop slutprodukten till en mindre klump.

Användarinstruktion

Vår COMpacker har allt man kräver av en modern charpacker. Den klarar långa filer som kan vara ända upp till cirka 244 blocks, d.v.s från $0801 till $F300. Detta borde tillfredsställa de flesta. Av denna anledning blev det väl svårt att dessutom lägga in en diskturbo, varför detta extra plus utelämnades.

Så här långa filer får man fram genom att spara ut minnet med ett anständigt cartridge, t.ex. Expert, Action Replay eller C.C.S. mon.

The Final Cartridge kan INTE generera denna typ av filer!

Då man skall packa sitt program ställs det ett antal frågor:

***"Filename"**

är namnet på det program man vill ladda. Anger man ett dollartecken ($) får man fram diskettens directory och denna "dir"-rutin har de vanliga finesserna; paus (space) och break (run/stop).

***"Savename"**

är det namn som den packade filen får på disketten. Man kan byta disk innan man sparar så det gör inget om filen redan skulle existera på den disk som innehåller källprogrammet.

***"Start"**

är startadressen på programmet angivet i hexadecimalform. Finessen här är att det blir en vanlig BASIC run om man inte anger någon startadress. Packar man sitt BASIC-program behöver man därför inte vara insatt i varken hexadecimal talnotering eller bankswitchning genom adress $01.

Man trycker i så fall bara på RETURN och vips har man en uppackningsrutin som kör på ett BASIC-program.

***"$01 Status"**

är något vi kan bortse från att förklara djupare eftersom frågan redan tagits upp grundligt i både juninumrets COMpostspalt och en av imponansartiklarna. Kort kan sägas att denna byte styr vilka ROM- och RAM- banker maskinen skall hämta data ur. Normalt anges här $37 och det skall vara annorlunda endast i undantagsfall.

Färdigpackat

Då skärmen delas upp i tre lika breda, randiga och lodräta fält vet man att det är dags att spara programmet, vilket för övrigt även talas om på skärmen. Som tidigare nämnt finns ingen turbo, då COMpackern även klarar av att det färdiga programmet är längre är 202 block. Då programmet är utsparat på disk får man frågan om man vill spara igen. Vill man inte detta är COMpackern redo att ta sig an nästa fil som skall packas.

Packoptimering

Innan man packar bör man komma ihåg de viktiga reglerna för bra packresultat: Bara aktiv kod och data skall vara med. Fyll minnet med nollor innan ni fixar ihop den fil som skall packas. Inget skräp bör finnas kvar, för det tar bara en massa onödig plats i den fördigpackade filen.

Kommentarer till listningen

Den programlistning som finns med i denna artikel är en s.k. BASIC-loader. Den omvandlar datasatserna i BASICen till en maskinkodsfil, vilken sparas ut på disk.

Resultatet på disk, "COMpacker 1.0/OC", är den fullt färdiga packern som är klar att börja användas. Kom ihåg att detta är ett Public Domain-program som får spridas och användas i sin OMODIFIERADE form.

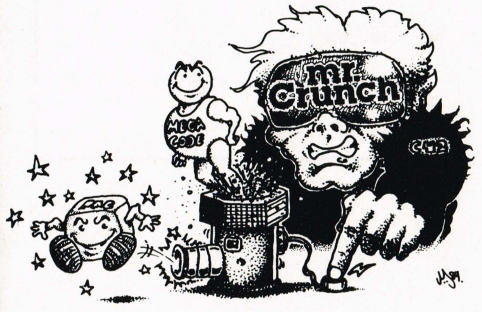
Eventuella synpunkter, positiva som negativa, belöningar, förslag och frågor kan skickas till redaktionen, så skall vi se om dessa kan beaktas i en framtida ny version.

Som avslutning skulle jag vilja passa på att sända ett hjärtligt tack till "Sjögullet" Rickard Sjögren för en strong programmeringsinsats.

v.1.0

Vad sägs om att få dina program att fylla mindre? COMputers COMpacker v.1.0 gör det hela möjligt, det är bara att knappa in!

Av Pontus Lindberg




```
10 I=24576
20 READ A:IF A=256 THEN SYS 24576
30 POKE I,A:I=I+1:GOTO 20
24576 DATA 169,98,133,76,169,96,133,
   77
24584 DATA 169,1,168,162,8,32,186,255
24592 DATA 169,16,162,82,160,96,32,
   189
24600 DATA 255,32,192,255,162,1,32,
   201
24608 DATA 255,169,1,32,210,255,169,8
24616 DATA 32,210,255,160,0,177,76,
   141
24624 DATA 32,208,32,210,255,230,76,
   208
24632 DATA 2,230,77,169,77,197,76,208
24640 DATA 234,169,101,197,77,208,
   228,169
24648 DATA 1,32,195,255,32,204,255,76
24656 DATA 226,252,67,79,77,80,65,67
24664 DATA 75,69,82,32,49,46,48,47
24672 DATA 79,67,11,8,97,79,158,50
24680 DATA 48,54,49,0,0,0,120,32
24688 DATA 129,255,32,132,255,32,138,
   255
24696 DATA 169,15,141,33,208,141,134,
   2
24704 DATA 32,68,229,169,0,141,32,208
24712 DATA 141,33,208,169,27,160,9,32
24720 DATA 30,171,162,2,160,12,24,32
24728 DATA 240,255,160,0,32,207,255,
   201
24736 DATA 13,240,15,201,36,208,3,76
24744 DATA 119,9,153,54,3,200,192,16
24752 DATA 208,234,140,52,3,162,3,160
24760 DATA 12,24,32,240,255,160,0,32
24768 DATA 207,255,201,13,240,8,153,
   70
24776 DATA 3,200,192,16,208,241,140,
   53
24784 DATA 3,162,4,160,12,24,32,240
24792 DATA 255,160,0,32,207,255,201,
   13
24800 DATA 240,11,153,86,3,200,192,4
24808 DATA 208,241,76,162,8,169,89,
   141
24816 DATA 179,12,169,166,141,180,12,
   169
24824 DATA 55,141,174,12,169,0,133,
   198
24832 DATA 76,83,10,173,86,3,32,240
24840 DATA 8,173,87,3,32,7,9,141
24848 DATA 180,12,173,88,3,32,240,8
24856 DATA 173,89,3,32,7,9,141,179
24864 DATA 12,169,0,32,210,255,162,5
24872 DATA 160,12,24,32,240,255,160,0
24880 DATA 32,207,255,201,13,240,239,
   153
24888 DATA 90,3,200,192,2,208,241,173
24896 DATA 90,3,32,240,8,173,91,3
24904 DATA 32,7,9,141,174,12,76,83
24912 DATA 10,41,63,201,7,176,11,24
24920 DATA 105,9,10,10,10,10,141,26
24928 DATA 9,96,56,233,48,76,249,8
24936 DATA 41,63,201,7,176,7,24,105
24944 DATA 9,13,26,9,96,56,233,48
24952 DATA 76,16,9,0,147,14,8,5
24960 DATA 99,111,109,80,65,67,75,69
24968 DATA 82,32,102,82,79,77,32,111
24976 DATA 66,69,82,79,69,78,68,69
24984 DATA 32,99,111,109,80,85,84,69
24992 DATA 82,13,13,102,73,76,69,78
25000 DATA 65,77,69,32,32,32,58,13
25008 DATA 115,65,86,69,78,65,77,69
25016 DATA 32,32,32,58,13,115,84,65
25024 DATA 82,84,32,32,32,32,32,32
25032 DATA 36,13,36,48,49,32,115,84
25040 DATA 65,84,85,83,32,36,13,0
25048 DATA 32,68,229,169,13,32,210,
   255
25056 DATA 169,1,162,8,160,0,32,186
25064 DATA 255,169,1,162,248,160,9,32
25072 DATA 189,255,32,192,255,162,1,
   32
25080 DATA 198,255,32,207,255,32,207,
   255
25088 DATA 32,207,255,32,207,255,240,
   58
25096 DATA 32,204,255,32,228,255,201,
   32
25104 DATA 208,5,32,228,255,240,251,
   201
25112 DATA 3,240,39,162,1,32,198,255
25120 DATA 32,207,255,170,32,207,255,
   32
25128 DATA 205,189,169,32,32,210,255,
   32
25136 DATA 207,255,240,6,32,210,255,
   76
25144 DATA 206,9,169,13,32,210,255,76
25152 DATA 159,9,169,1,32,195,255,32
25160 DATA 204,255,169,249,160,9,32,
   30
25168 DATA 171,32,228,255,240,251,76,
   13
25176 DATA 8,36,18,32,112,82,69,83
25184 DATA 83,32,65,78,89,32,75,69
25192 DATA 89,32,146,0,18,32,32,32
25200 DATA 32,108,79,65,68,73,78,71
25208 DATA 32,32,32,32,146,0,18,32
25216 DATA 115,80,65,67,69,32,116,79
25224 DATA 32,115,65,86,69,32,146,0
25232 DATA 18,32,32,32,32,115,97,118
25240 DATA 105,110,103,33,32,32,32,32
25248 DATA 146,0,18,115,97,118,101,32
25256 DATA 97,103,97,105,110,40,121,
   47
25264 DATA 110,41,146,0,162,7,160,2
25272 DATA 24,32,240,255,169,11,160,
   10
25280 DATA 32,30,171,162,0,169,1,157
25288 DATA 0,218,169,0,157,0,6,202
25296 DATA 208,243,142,14,220,162,
   236,160
25304 DATA 12,134,78,132,79,169,1,162
25312 DATA 8,160,0,32,186,255,173,52
25320 DATA 3,162,54,160,3,32,189,255
25328 DATA 32,192,255,162,1,32,198,
   255
25336 DATA 32,207,255,165,144,240,11,
   169
25344 DATA 1,32,195,255,32,204,255,76
25352 DATA 13,8,32,207,255,32,207,255
25360 DATA 160,0,230,1,145,78,198,1
25368 DATA 230,78,208,2,230,79,170,
   206
25376 DATA 32,208,254,0,6,208,3,222
25384 DATA 0,6,165,144,240,223,169,1
25392 DATA 32,195,255,32,204,255,162,
   0
25400 DATA 160,1,189,0,6,217,0,6
25408 DATA 176,6,200,240,8,76,217,10
25416 DATA 152,170,76,225,10,169,55,
   133
25424 DATA 1,88,142,143,12,142,154,12
25432 DATA 142,144,11,142,157,11,142,
   24
25440 DATA 11,162,7,160,2,24,32,240
25448 DATA 255,169,29,160,10,32,30,
   171
25456 DATA 169,0,141,14,221,238,32,
   208
25464 DATA 238,0,218,173,1,220,201,
   239
25472 DATA 208,243,162,7,160,2,24,32
25480 DATA 240,255,169,47,160,10,32,
   30
25488 DATA 171,169,1,141,65,12,162,8
25496 DATA 142,66,12,168,32,186,255,
   173
25504 DATA 53,3,162,70,160,3,32,189
25512 DATA 255,32,192,255,162,1,32,
   201
25520 DATA 255,169,12,133,77,169,65,
   133
25528 DATA 76,160,0,177,76,32,210,255
25536 DATA 230,76,208,2,230,77,169,
   236
25544 DATA 197,76,208,237,169,12,197,
   77
25552 DATA 208,231,120,169,56,133,1,
   32
25560 DATA 12,12,170,32,0,12,132,73
25568 DATA 192,3,144,25,152,24,101,76
25576 DATA 133,76,165,77,105,0,133,77
25584 DATA 169,0,32,241,11,165,73,32
25592 DATA 241,11,76,172,11,224,0,208
25600 DATA 12,138,32,241,11,169,0,32
25608 DATA 241,11,76,176,11,138,32,
   241
25616 DATA 11,165,76,197,78,208,187,
   165
25624 DATA 77,197,79,208,181,169,55,
   133
25632 DATA 1,169,1,32,195,255,32,204
25640 DATA 255,88,169,1,141,14,220,
   162
25648 DATA 7,160,2,24,32,240,255,169
25656 DATA 65,160,10,32,30,171,165,
   197
25664 DATA 201,25,208,3,76,0,11,201
25672 DATA 39,208,243,169,0,133,198,
   76
25680 DATA 13,8,198,1,88,32,210,255
25688 DATA 141,32,208,120,230,1,169,
   255
25696 DATA 96,160,0,209,76,208,5,200
25704 DATA 192,255,208,247,96,160,0,
   177
25712 DATA 76,230,76,208,2,230,77,96
25720 DATA 147,8,5,14,99,111,109,80
25728 DATA 65,67,75,69,82,32,118,49
25736 DATA 46,48,13,13,0,102,73,76
25744 DATA 69,78,65,77,69,58,0,13
25752 DATA 115,65,86,69,78,65,77,69
25760 DATA 58,0,1,8,11,8,103,18
25768 DATA 158,50,48,54,53,32,70,76
25776 DATA 84,0,0,0,160,96,120,230
25784 DATA 1,185,73,8,153,0,5,136
25792 DATA 16,247,169,170,56,229,45,
   133
25800 DATA 254,169,8,229,46,133,255,
   200
25808 DATA 198,46,206,54,8,177,45,153
25816 DATA 0,0,200,208,248,165,46,201
25824 DATA 8,176,237,230,46,169,1,133
25832 DATA 45,76,0,5,32,45,5,201
25840 DATA 0,208,77,32,45,5,201,0
25848 DATA 208,5,169,0,76,84,5,170
25856 DATA 32,45,5,32,66,5,165,255
25864 DATA 208,226,165,254,208,222,
   169,55
25872 DATA 133,1,88,32,89,166,76,174
25880 DATA 167,160,0,177,254,198,1,
   141
25888 DATA 32,208,140,32,208,230,1,
   230
25896 DATA 254,208,2,230,255,96,160,0
25904 DATA 145,45,230,45,208,2,230,46
25912 DATA 202,224,255,208,243,76,26,
   5
25920 DATA 160,0,145,45,230,45,208,2
25928 DATA 230,46,76,26,5,256
```

STORA SPRITES PÅ 64:AN

```
        *=$C000

        JSR SETVIC

        AKTIVERAR RASTERINTERUPT

        SEI
        LDA #$7F
        STA $DC0D
        LDA #1
        STA $D01A
        LDA #<IRQ
        LDX #>IRQ
        STA $0314
        STX $0315
        LDA #55
        STA $D012
        LDA #56
TIMEIN  CMP $D012
        BNE TIMEIN
        CLI
LOOP    JMP LOOP

;       INTERUPTRUTIN FÖR FÖRSTORING.
```

COMputers grafikexpert bjuder här på en fräck rutin till alla som gillar att leka med spritar. Häng med nu, för han har något STORT på gång!

Av Daniel Larsson

Många av er som vet hur spritegrafik fungerar på 64:an, har säkert någon gång tyckt att spritarna är alldeles för små. Normalt är de 24×21 pixels stora, och genom att ändra i addresserna $D017 och $D01D kan man göra dem dubbelt så stora i Y-led respektive X-led. Så är det med den saken, säger nog de flesta.

Då visste ni inte att man genom ett tämligen enkelt sätt kan få varje horisontell linje i en sprite hur hög (i Y-led, alltså) man vill.

Hur man förlänger en spriterad

Hur gör man då, undrar ni nu! Man använder adress $D017 (53271 decimalt) på ett lite anorlunda sätt. I princip går det till på följande sä't:

Plocka fram din sprite på skärmen, och förstora den i Y-led, genom att tända spritens motsvarande bit på adress $D017. Låt rasterstrålen rita upp den allra första linjen. På nästa rad ritas samma linje upp igen. Då sätter du adress $D017 till noll. På samma rasterrad förstorar du upp spriten igen. För sprite 0 kan man programmera det såhär:

```
LDA #00
STA #D017
LDA #01
STA #D017
```

Vad blir effekten då? Jo, när man sätter $D017 till noll, tror grafikchipet att den första av de två (varje spritelinje ritas upp två gånger när man har förstorat spriten i Y-led) linjerne håller på att ritas fram på skärmen. Då skall den alltså rita upp den andra (tror grafikchipet,ja!) linjen på nästa rasterrad. Om man upprepar det här mönstret på varje rad, så tror ju grafikchipet att det är samma linje som skall ritas upp hela tiden.

Nästa spriterad

Nu kommer problem nummer två: Hur skall man får grafikchipet att läsa in nästa spriterad? Det är enkelt! Man låter bli att förstora upp spriten igen, som exemplet ovan visar. Man sätter således adress $D017 till noll två gånger på samma rasterrad. Så här ser det ut i praktiken:

```
LDA #$00
STA #D017
LDA #$00
STA #D017
```

När den nya spritelinjen har lästs in, är det dags att förstora den det antal pixels som man vill ha den. Upprepa mönstret i det första programexemplet det antal linjer du vill ha den nyinlästa spriteraden, gör sedan som i exempel två.

Detta relativt enkla programmeringssätt (om än lite klurigt att greppa i början) möjliggör valfri förstoring av varje enskild spritelinje. Naturligt vis kan man förstora varje sprite individuellt, för adress $D017 innehåller ju varsin bit för alla åtta spritar.

Ett annat användningsområde

En del av er kanske vet som menas med 'DYSP'. 'DYSP' står för 'Different Y Sprite Positioning'. Dysp används när man vill röra på sprites i ramen. När man plockar bort ramen, är timingen MYCKET kritisk. Timingen beror bland annat på antalet sprites man har på varje rasterrad. Ifall man då flytta på spritarna i Y-led fungerar inte rutinen som tar bort ramen, eftersom processorn har olika antal fria

```
IRQ     NOP             ;TIMING
        BIT $FF         ;TIMING
        LDY #0
        LDX #0
ALLIN   DEC $D016       ;<-- TAR BORT
        INC $D016       ;<-- RAMEN
        LDA YTAB,X      ;TAR BORT TECKENPEKARE, ANNARS
        STA $D011       ;BLIR TIMINGEN FEL.
        STY $D017       ;<-- NOLLSTÄLL $D017 !
        LDA $C900,X     ;<-- FÖRSTORA
        STA $D017       ;<-- IGEN.
        NOP             ;TIMING
        NOP             ;TIMING
        BIT $FF         ;TIMING
        INX
        CPX #140        ;<-- 140 LINJER
        BNE ALLIN

        LSR $D019
        LDA #25
        STA $D011
        LDA #$FF
        STA $D017
        JSR LARGE
        LDA #55         ;<-- INTERUPTET SKALL VARA
        STA $D012       ;<-- PÅ RASTERRAD 55
        JMP $EA81       ;AVSLUTA INTERUPT
```

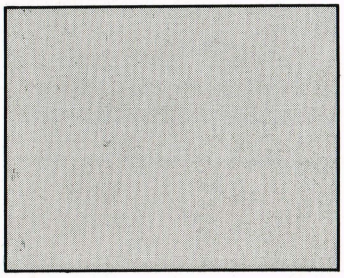
```
;       DENNA RUTIN LÄGGER IN DE RÄTTA
;       VÄRDENA PÅ ADRESS $C900 OCH
;       FRAMÅT. ANROPAS VID VARJE
;       INTERUPT. (SE OVAN I IRQ-RUTINEN!)

LARGE   LDX #0
        LDA #%11111111
CLR     STA $C900,X
        INX
        CPX #140
        BNE CLR
NUTHIN  LDX INPEK
ALLAX   LDA FLEX,X
        CLC
        ADC INUTI
        STA INUTI
        TAY
        LDA $C900,Y
ANDEN   AND #%11111110
        STA $C900,Y
        INC INUTI
        INX
        DEC RADER
        BNE ALLAX
        LDA #0
        STA INUTI
        LDA #20
        STA RADER
        LDA INPEK
        CLC
        ADC #10
        STA INPEK
        SEC
        ROL ANDEN+1
        LDA ANDEN+1
        CMP #$FF
        BNE NUTHIN
        LDA #0
        STA INUTI
        LDA #20
        STA RADER
        LDA #$FE
        STA ANDEN+1
FLT     LDA #0
        STA INPEK
        INC FLT+1
        RTS
INPEK   .BYT 0
INUTI   .BYT 0
RADER   .BYT 21

;       FÖRBEREDANDE RUTINER,
;       T.EX. FRAMPLOCKNING AV SPRITAR.
```

```
SETVIC  LDA #$FF
        STA 53269
        STA 53277
        LDX #0
        STX 53280
        STX 53281

ALLX    LDA POS,X
        STA 53248,X
        INX
        CPX #$12
        BNE ALLX
        LDA #13
        STA 2040
        STA 2041
        STA 2042
        STA 2043
        STA 2044
        STA 2045
        STA 2046
        STA 2047
        LDX #0
        STX $3FFF

FILL    TXA
        STA 832,X
        INX
        CPX #63
        BNE FILL
        LDX #0
        STX 832
        STX 833
        STX 834
        STX 892
        STX 893
        STX 894


FILLY   LDA #26
        STA YTAB,X
        LDA #27
        STA YTAB+1,X
        LDA #28
        STA YTAB+2,X
        LDA #29
        STA YTAB+3,X
        LDA #30
        STA YTAB+4,X
        LDA #31
        STA YTAB+5,X
        LDA #24
        STA YTAB+6,X
        LDA #25
        STA YTAB+7,X
        TXA
        CLC
        ADC #8
        TAX
        BNE FILLY


COPY    LDA FLEX,X
        STA FLEX+256,X
        INX
        BNE COPY
```

```
        LDA #8
        CLC
COLOUR  STA 53287,X
        ADC #1
        INX
        CPX #8
        BNE COLOUR
        LDX #0
PRINT   LDA TEXT,X
        BEQ BAGE
        JSR $FFD2
        INX
        BNE PRINT
BAGE    RTS

;       TECKENUTSKRIFT

TEXT    .BYT 147,5,8,142,13,13,13
        .BYT '          OBEROENDE COMPUTER 1989'
        .BYT 0


;       SPRITEPOSITIONER

POS     .BYT $00,$37,$30,$37
        .BYT $60,$37,$90,$37
        .BYT $C0,$37,$F0,$37
        .BYT $20,$37,$50,$37
        .BYT %11000000      ;       $D010
        .BYT $1B            ;       $D011


;       TABELL FÖR FÖRSTORING.
;       0= 1 LINJE HÖG
;       1= 2 LINJER HÖG
;       2= 3 LINJER HÖG
;       ...ETC...

FLEX    .BYT 0,0,0,0,0,0,0,0
        .BYT 1,1,1,2,2,2,3,3
        .BYT 3,4,4,4,5,5,5,6
        .BYT 6,6,7,7,7,6,5,4
        .BYT 3,2,1,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 1,1,2,2,3,3,4,4
        .BYT 5,5,6,6,7,7,8,8
        .BYT 9,8,7,6,5,4,3,2
        .BYT 1,0,0,0,0,0,0,0
        .BYT 1,1,1,2,2,2,3,3
        .BYT 3,4,4,4,5,5,5,6
        .BYT 6,6,6,6,6,6,6,6
        .BYT 6,6,6,6,6,6,6,6
        .BYT 5,5,5,5,4,4,4,4
        .BYT 3,3,3,3,2,2,2,2
        .BYT 1,1,1,1,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
```

```
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0
        .BYT 0,0,0,0,0,0,0,0

YTAB    =$C500
```

STORA SPRITES PÅ 64:AN



klockcykler för varje enskild linje. I stället för att med t.ex. NMI-timers göra så att man kan placera spritarna var man vill, kan man använda sig av flexibel förstoring. Placera alla spritarna på understa raden i varje sprite. Om du ritar de två nämnda linjerna tomma, så att de inte syns, blir effekten att spritarna rör sig enskilt i Y-led, i stället för att förstoras upp.

Programexemplet

Jag har gjort ett programexempel, som på ett enkelt sett visar hur effektfullt det ser ut när man töjer ut spritarna med hjälp av $D017. I assemblerlistningen finns korta förklaringar till programmet. Om ni vill ändra på utseendet när spritarna gungar, ändra då helt enkelt i raderna som jag har markerat med 'FLEX'. I basic-programmet motsvarar det rad 750 och framåt. Programmet tar också bort ramen, för att illustrera vad jag menade när jag sade att man kan använda rutinen till 'DYSP'. Om ni tycker att spritarna ser konstiga ut, har ni inte knappat fel. Det beror på att om jag skulle lägga in spritedata i programmet, skulle det ta upp 64×8=512 bytes mer. Den programmeringskunninge kan enkelt definiera spritarnas utseende själv. Har du ingen assembler, eller maskinkodsmonitor, så knappa in basicprogrammet. Det är värt all möda, det lovar jag, för resultatet ser verkligen häftigt ut!

```
100 REM ****************************
    ***
110 REM ***       FLEXY SPRITES
    ***
120 REM ***     AV DANIEL LARSSON
    ***
130 REM ****************************
    ***
140 :
150 :
160 CLR
170 READ A%:IF A%<>999 THEN POKE 491
    52+I%,A%:C=C+A%:I%=I%+1:GOTO 170
180 IF C<>54503 THEN PRINT"FEL I DAT
    ASATSERNA!":END
190 FOR X=0 TO 255:POKE 49152+I%+X,0
    :NEXT
200 PRINT"SKRIV 'SYS 49152' FöR ATT
    STARTA.        GLöM EJ ATT SPARA P
    ROGRAMMET!"
210 DATA 032,201,192,120,169,127,141,
    013
220 DATA 220,169,001,141,026,208,169,
    040
230 DATA 162,192,141,020,003,142,021,
    003
240 DATA 169,055,141,018,208,169,056,
    205
250 DATA 018,208,208,251,088,076,037,
    192
260 DATA 234,036,255,160,000,162,000,
    206
270 DATA 022,208,238,022,208,189,000,
    197
280 DATA 141,017,208,140,023,208,189,
    000
290 DATA 201,141,023,208,234,234,036,
    255
300 DATA 232,224,140,208,226,078,025,
    208
310 DATA 169,025,141,017,208,169,255,
    141
320 DATA 023,208,032,101,192,169,055,
    141
330 DATA 018,208,076,129,234,162,000,
    169
340 DATA 255,157,000,201,232,224,140,
    208
350 DATA 248,174,198,192,189,173,193,
    024
360 DATA 109,199,192,141,199,192,168,
    185
370 DATA 000,201,041,254,153,000,201,
    238
380 DATA 199,192,232,206,200,192,208,
    228
390 DATA 169,000,141,199,192,169,020,
    141
400 DATA 200,192,173,198,192,024,105,
    010
410 DATA 141,198,192,056,046,131,192,
    173
420 DATA 131,192,201,255,208,195,169,
    000
430 DATA 141,199,192,169,020,141,200,
    192
440 DATA 169,254,141,131,192,169,000,
    141
450 DATA 198,192,238,190,192,096,000,
    000
460 DATA 021,169,255,141,021,208,141,
    029
470 DATA 208,162,000,142,032,208,142,
    033
480 DATA 208,189,155,193,157,000,208,
    232
490 DATA 224,018,208,245,169,013,141,
    248
500 DATA 007,141,249,007,141,250,007,
    141
510 DATA 251,007,141,252,007,141,253,
    007
520 DATA 141,254,007,141,255,007,162,
    000
530 DATA 142,255,063,138,157,064,003,
    232
540 DATA 224,063,208,247,162,000,142,
    064
550 DATA 003,142,065,003,142,066,003,
    142
560 DATA 124,003,142,125,003,142,126,
    003
570 DATA 169,026,157,000,197,169,027,
    157
580 DATA 001,197,169,028,157,002,197,
    169
590 DATA 029,157,003,197,169,030,157,
    004
600 DATA 197,169,031,157,005,197,169,
    024
610 DATA 157,006,197,169,025,157,007,
    197
620 DATA 138,024,105,008,170,208,209,
    189
630 DATA 173,193,157,173,194,232,208,
    247
640 DATA 169,008,024,157,039,208,105,
    001
650 DATA 232,224,008,208,246,162,000,
    189
660 DATA 115,193,240,006,032,210,255,
    232
670 DATA 208,245,096,147,005,008,142,
    013
680 DATA 013,013,032,032,032,032,032,
    032
690 DATA 032,032,032,079,066,069,082,
    079
700 DATA 069,078,068,069,032,067,079,
    077
710 DATA 080,085,084,069,082,032,049,
    057
720 DATA 056,057,000,000,055,048,055,
    096
730 DATA 055,144,055,192,055,240,055,
    032
740 DATA 055,080,055,192,027,000,000,
    000
750 DATA 000,000,000,000,000,001,001,
    001
760 DATA 002,002,002,003,003,003,004,
    004
770 DATA 004,005,005,005,006,006,006,
    007
780 DATA 007,007,006,005,004,003,002,
    001
790 DATA 000,000,000,000,000,000,000,
    000
800 DATA 000,000,000,000,000,000,000,
    000
810 DATA 000,000,000,000,000,000,000,
    000
820 DATA 000,000,000,000,000,001,001,
    002
830 DATA 002,003,003,004,004,005,005,
    006
840 DATA 006,007,007,008,008,009,008,
    007
850 DATA 006,005,004,003,002,001,000,
    000
860 DATA 000,000,000,000,000,001,001,
    001
870 DATA 002,002,002,003,003,003,004,
    004
880 DATA 004,005,005,005,006,006,006,
    006
890 DATA 006,006,006,006,006,006,006,
    006
900 DATA 006,006,006,006,006,005,005,
    005
910 DATA 005,004,004,004,004,003,003,
    003
920 DATA 003,002,002,002,002,001,001,
    001
930 DATA 001,000,000,000,000,000,000,
    000
940 DATA 999
```

AMIGA MAGASINET

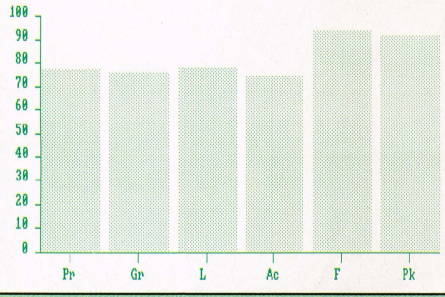
Innehållsförteckning:

Amigames	26
IT'S FILMTAJM, FOLKS!	30
AmigaBASIC	34
CLI-skolan, 3	36
Snabbisar	39
DOS-2-DOS	40

AMIGAMES

PALADIN

Presentation	78%
Grafik	76%
Ljud	79%
Action	73%
Fängslande	93%
Pris/kvalitet	91%



Redan från starten stöter du på förbittrat motstånd i Paladin. Du lär dock snabbt att se sammahanget i spelet, vilket gör det betydligt lättare att överskåda det hela.

I gengäld stiger svårighetsgraden konstant, vilket medför att du hela tiden får spänning i spelet.

Har du spelat Telewar kommer du att nicka igenkännande åt de strategiska principerna i Paladin. Även om det här är krigare och trollkarlar du flyttar runt på istället för stridsvagnar, är det fortfarande räckvidden i movement pionts som är det centrala elementet för hur äventyret ska gå. Tyvärr kan du bara spela mot datorn, men då den spelar tämligen intelligent bjuder den ändå på ordentligt motstånd.

Grafiken är om inte suverän så i varje fall tillräckligt fascinerande att hålla intresset uppe tills du utforskat vartenda rum i hela spelet.

Dessutom ska nämnas att Paladin är det enda spelet denna månad som utan problem kan läggas på hårddisk. Mycket användarvänligt.

Claus

I spelet Paladin ska du som en annan helig krigare bege dig ut på ett antal quests, det ena mer farofyllt än det andra. Spelet har många D&D-liknande ingredienser, men det primära i spelets idé är ändå att du ska ha en god stategisk förståelse för hur man bygger upp ett angrepp.

Du startar varje quest med att råda över ett varierande antal svärdfäktare, magiker samt en enda paladin. Varje man i din grupp har sin egen personlighet i form av prestation, rörelser och hälsa.

Movement points spelar en rätt central roll i Paladin, då de utöver att ange hur långt du kan flytta med dina män även bestämmer hur många svärdhugg, pilar och spells du kan skicka iväg per omgång. Det är bara magikerna som kan skicka iväg spells, vilket gör dem tämligen viktiga.

Dina svärdbärare är förutom svärd utrustade med pilbågar så att de även kan angripa motståndare de inte har tillräckligt med movement points att komma ända fram till.

Det finns givetvis en hel massa quests att ge sig ut på i spelet, men tröttnar du på dem kan du göra egna äventyr med hjälp av det medföljande quest design- programmet.

När Claus förlorade lottningen om vem som skulle recensera Paladin skrattade jag i smyg, jag hade nämligen sett vilken jättemanual han skulle bli tvungen att tröska igenom. Men när han väl började spela blev jag så gripen att jag snodde manualen för honom. För spelet är verkligt spännande, och det krävs en hel del strategi för att undvika att alla ens män utrotas på fem minuter i en orgie av "orbs of fire"!

När jag väl nackat alla fiende som skulle nackas och bara hade en enda kvar, fyrade jag lugnt av en fireball med det resultat att jag stekte ming själv och plötsligt hade förlorat det spel jag varit så säker på att vinna. Detta hade Claus mycket roligt åt, så jag hämnades genom att sabba hans väckarklocka.

De många olika questsen gör att spelet inte blir tråkigt ens efter lång tid, och då spelet dessutom är mycket enkelt att styra är det verkligen en fröjd att spela det.

Vi får ge oss på nåd och onåd och erkänna att Paladin är ett verkligt toppspel som förtjänar en plats hos envar spelfreak.

Hans

CASTLE WARRIOR

Det var en gång en liten idyllisk diktatur, ursäkta - kungarike, där den högt älskade kungen Edelwulf regerade med en sällan skådad rättfärdighet. Hans motståndare, den elake trollkarlen Zandor, hade han deporterat till en gudsförgäten utpost långt, långt bort. Men tydligen inte tillräckligt långt, för Zandor lyckas ändå med bedriften att förgifta Edelwulfs mat.

Ve och ack! Ett motgift måste skaffas fram så snabbt som möjligt, och uppgiften att hitta detta motgift faller på dig. Du ska in i Zandors slott och stjäla motgiftet, en uppgift som inte är helt problemfri.

Runt i slottets mörka gångar ska du vandra under ständig attack av fåglar och armar som sträcks ut från väggarna för att gripa dig. Ovanligt nog går du hela tiden inåt i en 3D-skärm, du ser alltså hela tiden bara ryggen av din krigare.

THE NEWZEALAND STORY

Vi lämnar ännu en gång ordet till vår special-korrespondent för supergulliga spel, för det är knappast någon annan som kan klara av att beskriva Oceans konvertering av spelmaskinen "The New Zealand Story".

Här spelar du rollen som en kiwi (som dels är en fågel, dels en frukt och dels australisk slang för nyazealändare), som bor i ett stilla och fredligt zoo. Fredligt är det ända tills den kiwimåltidsälskande valrossen dyker upp och kidnappar alla dina vänner. De ska naturligtvis räddas från den stora grytan, så det är bara att ge sig ut i den vilda jakten över 20 vitt skilda banor.

På din väg ska du använda dig av din dykutrustning samt ballonger och andra flygredskap. Dessutom flyger vissa fiender runt på något som liknar ekollon, och kan du putta undan en sådan fiende har du också snott åt dig ett transportmedel.

I slutet av varje bana ser du en bur med en av dina vänner i, men det händer att där även finns en vakt som du måste tala tillrätta först. Detta kräver olika tekniker, t ex ska du likt Jona låta dig slukas av valen för att kunna skjuta den inifrån.

Normalt kopplar man ihop arkadspel med en våldsam handling återgiven i en orgie av imponerande explosioner. The New Zwaland Story är defintivt ett undantag. Spelet är utfört i en supergullig grafik som är något av det mest humorisktiska jag sett på länge i ett arkadspel.

Jag måste medge att jag blev en smula imponerad av alla de möjligheter som erbjuds dig som spelare. Du kan t ex använda dig av många av de farkoster som dina fiender använder. Dessutom kan du dyka, vilket ser rätt kul ut när din kiwi tar på sig cyklop och snorkel.

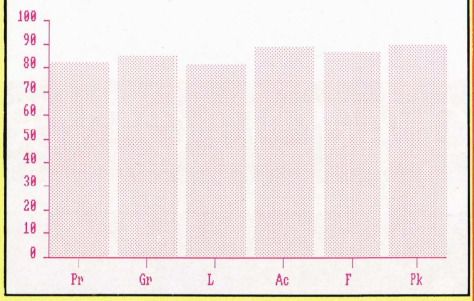
Många kommer säkert och skaka på huvudet när de ser New Zealand Story, och de har också helt rätt i att spelet inte är helt normalt - men det är just det som är charmen med spelet.

Banorna är gjorda så att du nästan alltid ramlar i en massa fällor första gången du provar dem. Man lär sig dock snabbt att undvika dessa, så det dröjer ett bra tag innan man fastnar totalt i spelet.

The New Zealand Story är både kul, annorlunda, humoristiskt och fängslande, vilket är sällsynt att stöta på i ett och samma spel. Det enda jag egentligen saknar är en tvåspelarmöjlighet där man kan hjälpa varandra i spelet.

Claus

Presentation	82%
Grafik	85%
Ljud	81%
Action	99%
Fängslande	86%
Pris/kvalitet	89%

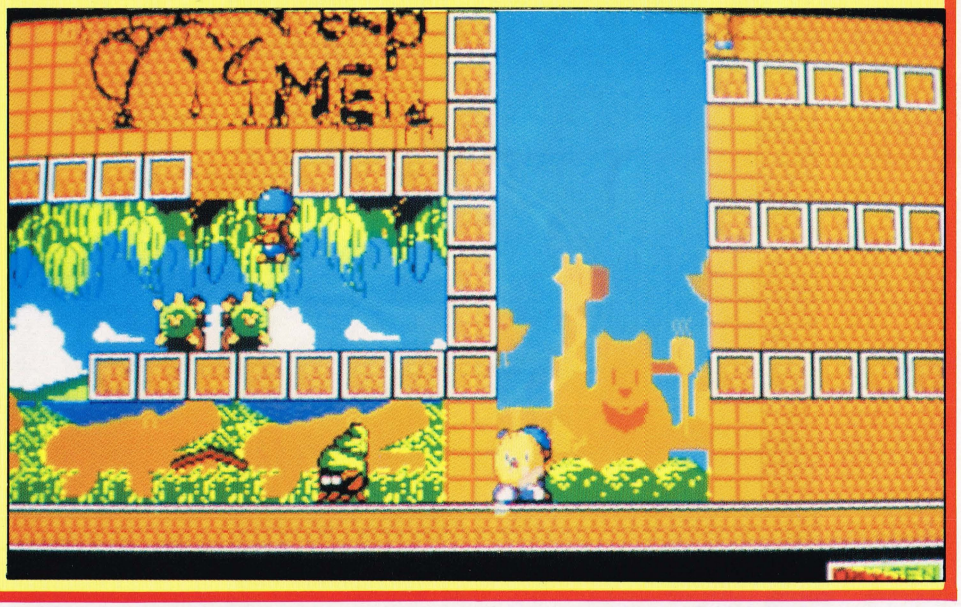


När jag laddade in The Newzealand Story första gången hade jag precis innan spelat Barbarian II och dreglat över presentationen, och den syn som nu mötte mig påminde mest om något jag en gång sett Claus rita på en Spectrum. Senare blir dock presentationen bättre med en scen där valrossen kidnappar kiwikycklingarna, och när jag väl startade på själva spelet blev jag strax helsåld på de små söta spritarna som far runt på skärmen.

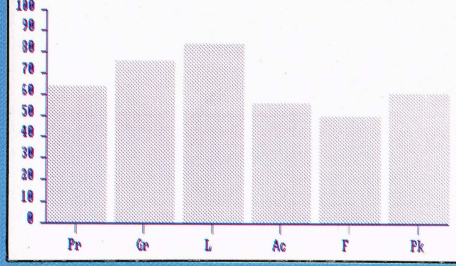
Spelets koncept är precis som det ska vara, och här finns verkligen arkadkänsla så det smäller om det. Banorna är precis så olika att man alltid har lust att se en till och melodin som kör i bakgrunden är så sockersöt att den utan tvivel kommer att bli huvudfavorit i nästa års schlagerfestival.

Det enda jag egentligen saknar är att det inte kommer ut en back kalla pilsner genom högtalaren när man genomfört spelet. Kan inte programmerarna ta sig samman lite och skriva några spel som är både underhållande och nyttiga?

Hans



Presentation	64%
Grafik	77%
Ljud	84%
Action	56%
Fängslande	49%
Pris/kvalitet	61%



"Wow" var min första kommentar till Castle Warrior när jag såg bilderna på förpackningens baksida. "Uuurk" var min nästa kommentar när jag började spela själva spelet. Visserligen är grafiken ganska flott och animationen med många frames är heller inte att förakta, men det är verkligt irriterande att styra figuren på skärmen.

Det tar i regel en halv sekund från det man ber honom om att utföra en rörelse tills det rent faktiskt händer något. Det låter kanske inte så illa, men när man samtidigt måste ha en exakt timing på sina slag för att träffa inser du lätt hur frustrerande det är.

Och det blir inte bättre när du kommer fram till de avslutande fienderna. Här ska du reflektera deras eldkulor, men då man inte får någon känsla alls för var kulorna befinner sig blir det mest slag i luften tills man träffar av en ren slump.

Ljudet är hyfsat, så efter ett tag vände jag ryggen till spelet för att kunna lyssna till låten utan att behöva irriteras av spelet.

Hans

Det är svårt att inte bli en smula irriterad på Castle Warrior. Perspektivet ställer till problem, och det blir inte bättre av att din krigare "skymmer sikten". Men övning ger resultat och det visar sig så småningom att det inte är särskilt svårt (eller spännande) att komma långt i spelet.

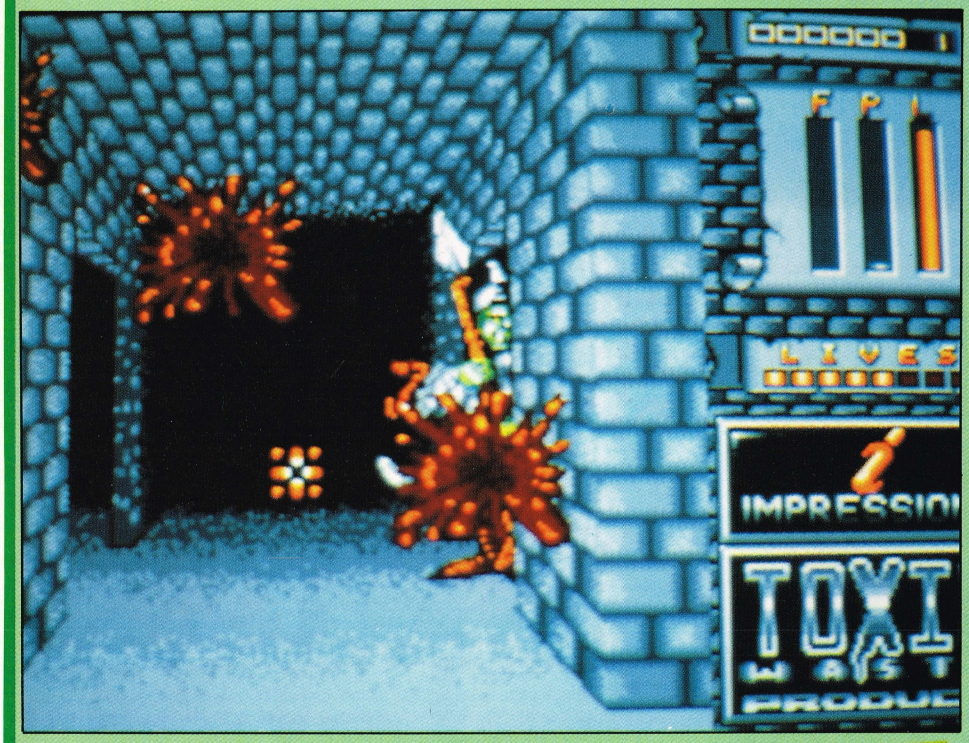
Grafiken i Castle Warrior är rätt dekorativ, men väggarna som kör förbi påminner starkt om en rulltrappa till utseendet och när du träffar en fågel eller en arm stannar spelet upp ett ögonblick för att göra en explosion. Det känns som om alla rörelser i spelet är en smula klumpiga.

Jag måste tyvärr göra Delphine Software besviken genom att konstatera att Castle Warrior saknar precis de ingredienser som hade gjort spelet fängslande. Idén kunde varit bättre, och programmeringen, här speciellt grafikbehandlingen, är faktiskt under all kritik.

Claus

AMIGAMES

CHARIOTS OF WRATH



Här kommer ett spel i form av en tårta. Sex olika småspel ligger på varandra och det hela är överdraget med lite förhistorieglasyr.

Du spelar rollen som en prins som är så brutal att kungen inte låter dig följa med när det ska ut ock krigas. När han får veta att hans älskade Arne - trams: Arthena har kidnappats av den elake baronen, och då är det ju inte annat att göra än att försöka giva honom på moppo.

På första banan ser du 2 dörrar bakom vilka det dyker upp diverse orörliga monster som du har ett par sekunder på dig att träffa. Nästa bana är ett breakoutspel à la Arkanoid, dock väsentligt förenklat. Sedan är det dags för lite rymdspel i stil med Hybris, men återigen på en mycket låg nivå. Nu tror du kanske att det dyker upp ett nytt spel. Icke, du ska tillbaka till breakouten igen, och först efter detta får du lov att hoppa runt på plattformar och samla escape pods till din vidare färd. Det finns inte en fiende inom mils omkrets, så om du inte drattar omkull på egen hand är det inte farligare än att gå en sväng på stan en söndagförmiddag.

Så fortsätter det med ännu ett rymdspel innan du äntligen kommer fram och kan rädda Arne.

Tja, det ser ut som om softwarefirman Impressions i och med Chariots of Wrath försökt sig på att få fram ett spel som har allt. Breakout, rymdspel och små gubbar som hoppar på plattformar.

Tyvärr blir resultatet att vad spelet har i kvantitet saknar det i kvalitet. Tar man varje avdelning separat är Chariots of Wrath nämligen ett verkligt svagt spel. Breakoutavdelningen kunde enkelt gjorts avsevärt bättre, och om man jämför med något av de existerande breakoutspelen på marknaden blir jämförelsen nästan pinsam för Chariots.

Och så är det hela vägen.

Grafiken i Chariots of Wrath är egentligen ganska flott, men vad hjälper det när spelet sänks totalt av usla programmeringsrutiner.

Claus

Jag har aldrig hört talas om spelfirman Impressions förr, och efter att ha sett vad de presterat i Chariots of Wrath har jag heller ingen lust att höra talas om dem igen. Den programmeringsgrupp som stått för själva programmeringen heter Toxic Waste (giftigt avfall), och det måste sägas vara ytterst betecknande för vad de producerat här.

Den första banan lider av en total avsaknad av animation och jag tror faktiskt att jag kunde yxat ihop något motsvarande i BASIC på ett par timmar! Innan nästa bana står dett att man ska trycka space för att fortsätta, och det säger lite om spelets kvalitetsnivå att man kan hamra på mellanrumstangenten i flera timmar om man vill, det ändrar inte på det faktum att man måste trycka fire för att få lov att spela det extremt simpla breakoutspelet.

Här kommer bokstäver flygande precis som i Arkanoid så att man t ex kan få två kulor, men då dessa avfyras precis samtidigt mot var sin ände av banan och man nu bara har en bat kan man inte använda dem till något.

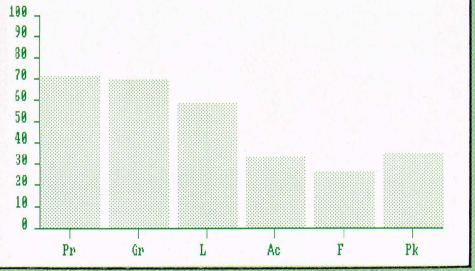
Rymdspelet är lite bättre, men i gengäld så lätt att till och med min mormor kunde klarat det i första försöket med förbundna ögon inne i en cementblandare under häftig maskingevärsbeskjutning.

Det är helt enkelt så lätt att det blir tråkigt.

Det är en mycket god idé att lägga sex olika spel i ett, men när inget av dem är tillräckligt intressant för att spelas separat faller hela idén platt till marken. Kort sagt, Frankie says: köp inte det här spelet, det är förbjudet för barn över 4 år.

Hans

Presentation	71%
Grafik	70%
Ljud	58%
Action	33%
Fängslande	26%
Pris/kvalitet	34%



BARBARIAN II

Det lär väl inte vara många Amigaägare som inte stiftat bekantskap med Barbarian - spelet där det vimlade med avhuggna huvuden och blodiga svärd. Detta faktum har inte undgått Palace Software, och vad är då naturligare för dem än att lansera Barbarian II som i blodighet är vida överlägset sin föregångare.

Här har den onde Drax förskansat sig i sin borg och hotar med att förstöra hela kungariket och halva prinsessan Marina. Därför ska The Barbarian eller Marina (välj själv) kämpa sig igenom tre olika nivåer för att nå in till The Sanctum of Drax. Där ska du först ta hand om fiendens män och sedan så småningom stå öga mot öga mot Drax himself.

Men först är det alltså de där tre nivåerna som var för sig är befolkade med sex typer odjur. Alla slåss på sitt eget sätt, från apan som sparkar mot dig till Saurion Beasts som knipsar skallen av dig med ett enda bett om du inte håller dem på avstånd.

Varje bana är en labyrint som består av ca 28 skärmar som inbördes är sammankopplade med portar. Bilden ses inte alltid från samma håll, så för att du ska ha en chans att hitta runt finns ett svärd nederst på skärmen som alltid pekar mot norr. Visst har man hjälp av det, men man kommer knappast igenom utan att rita en karta. Det är så att man ska hitta två magiska saker på varje bana, och då dessa skiftar position från spel till spel blir man så illa tvungen att söka igenom alla rum.

När du slåss har du möjlighet till tre olika slag och en spark, ju svårare det ör att utföra desto mer skada vållar det. Bäst är "Flying Neck Chop" som är mycket svårt att träffa med men i gengäld som regel resulterar i ett avhugget huvud när det lyckas.

Förutom monstren finns det också en del hål i banorna. Dessa hål kan vara mycket svåra att undvika att ramla i när ett monster tränger dig bakåt.

Redan uppstartssekvensen i Barbarian II visar att man jobbat en hel del med spelet. Det vidriga kranium som dyker upp och med samplad röst talar om att du ska skifta disk samtidigt som det viftar med sin skeletthand är helt enkelt suveränt gjort, både grafik- och ljudmässigt.

Ofta ser man att ett spel inte riktigt kan leva upp till presentationen, men så är inte fallet här. På första skärmen hör du ett flott vindsus i stereo samtidigt som du ser ett huvud vars hår fladdrar i takt med ljudet spetsat på en påle.

När du sedan provar dina olika manövrar blir du inte mindre positivt överraskad, för animationen är helt i topp med massor med olika frames (bilder). Det enda man kan rikta en anmärkning mot är att själva teckningarna kunde varit bättre.

Gameplayet är verkligt fint och de olika monstren slåss verkligen på helt skilda sätt. Detta gör att spelet blir verkligt omväxlande. Jag har faktiskt sällan dreglat över så mycket blod och hugga av armar och... (lugn Hans, tänk på att du skriver för helt oskyldiga läsare)... och sitta och njuta med ett stilla glas mjölk en söndag eftermiddag.

Hans

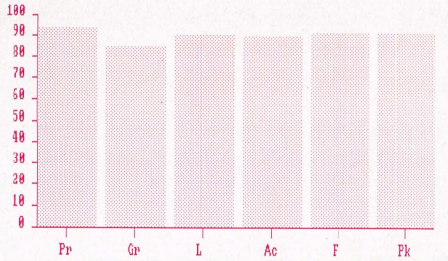
Barbarian II till C64 var och är ett riktigt kvalitetsspel fyllt med en hel massa dråpliga och bloddrypande grafikeffekter. Bortsett från en helt suverän intro är skillnaden mellan C64- och Amigaversionen svår att hitta. Amigaversionen har en något högre upplösning, men när det gäller handling och omgivningar är de två versionerna faktiskt fullständigt identiska.

Så höll jag ju nästan på att glömma ljudet i spelet. Här innebär Amigaversionen en väsentlig förbättring. Samplade svärdhugg och stönanden i stereo är finesser som gör att det helt klart lönar sig att ansluta spelet till en stereo (eller få en att önska att man hade en bättre).

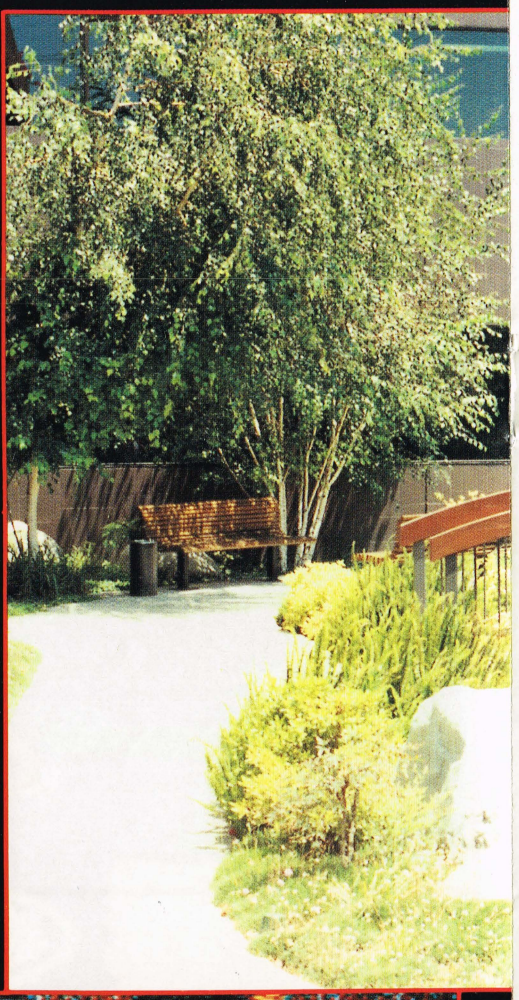
Barbarian II är knappast det mest suveräna kampspel vi sett på en Amiga, men det råder inga tvivel om att det hör till toppklassen att att det ligger många timmars fängslande underhållning framför alla som anser sig kunna hantera en joystick med viss fermitet.

Claus

Presentation	94%
Grafik	85%
Ljud	90%
Action	89%
Fängslande	91%
Pris/kvalitet	91%



IT'S FILMTAJM, FOLKS!



Cinemaware ligger ordentligt undangömt vid sidan av allfarvägen utanför L.A. mitt i en klunga av high-tech-firmor.

COMputer har besökt en spelfirma, där direktören tror att han egentligen gör film. Jodå, det är givetvis Cinemaware vi hälsat på hos.

Av Morten Strunge Nielsen

Introbild till "TV Sports Basketball".

Cinemaware bildades i januari 1986 i solstänkta Californien under namnet Master Designer Software Incorporated. I den korta, men ytterligt precisa firmabeskrivningen, står det bara att "firmans avsikt är att utveckla och marknadsföra en ny, innovativ serie underhållningssoftware vid namn Cinemaware".

I november 1987 tog Bob Jacobs så steget fullt ut och döpte om hela firman till det mer välklingande Cinemaware Corporation.

Det började med en Amiga

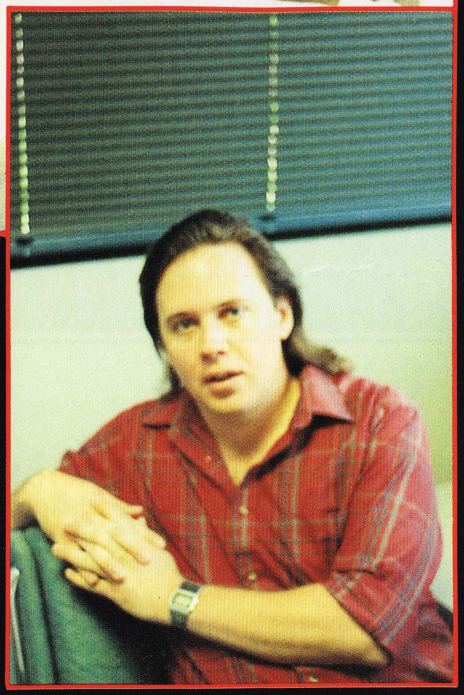
I starten var det bara Phyllis och Bob Jacobs, och kontoret var ett litet gästrum i deras villa. Det dröjde dock inte länge innan verksamheten växte dem ur huset. Nu, drygt tre år senare, räknar personalstyrkan 70 själar.

Men nu var ju herr och fru Jacobs inte vilka som helst heller...

Phyllis arbetade hos en agent som bl a räknade in Charles Bronson, John Huston och andra kända skådespelare bland sina klienter. Bob var direktör för Next Frontier Corporation, en firma som sysslade med att licensiera ut software till de kända spelfirmorna.

Bob pysslade också lite med programmering på sin fritid, program som såldes vidare till Activision och Aegis Development. Han utvecklade dessutom Rapidlok64, ett kopieringsskydd till Commodore 64 som han sålde till Accolade, Microprose och många andra.

Man kan säga att det hela startade med två lyckliga slumpar. Först blev Bob föräl-



David Todd jobbade för NASA innan han kom till Cinemaware.

skad i supermaskinen Amiga, dessutom träffade han speldesigneren Kellyn Beek hos Epyx. Kellyn hade massor med goda idéer, och det var givetvis han som fick lov att designa Defender of the Crown (och senare Rocket Ranger).

Cinemaware skulle nämligen inte bara producera enkla actionspel till Amiga med förbättrat ljud och grafik. Ne, de skulle göra "interaktiva filmer", dvs spel inspirerade av film så att Amigans suveräna ljud- och bildmöjligheter utnyttjades till fullo. Spel som appellerar även till en något äldre publik vars reaktionshastighet inte längre är på toppen.

Och Cinemaware gör verkligen allt för att deras spel ska påminna om spelfilmer. Disketterna kallas för rullar, spelen innehåller alltid production credits och de tidigaste spelens förpackningar var t o m utförmade som popcornaskar! Sist men inte minst vinnägger sig Cinemawarefolket om att få fram samma grafiktekniker som på film.

"Jo, jag vet att det är datorspel vi tillverkar, men vi försöker ändå marknadsföra och distribuera dem som film", säger Bob Jacobs.

Åtta utgivningar klokare

Men redan 1986 krävdes det gott om slantar för att sätta hjulen i rullning. Även här lånade Bob ideer från Hollywood. Han ordnade fram riskvilligt kapital genom "begränsat delägarskap". Och Cinemaware lyckades faktiskt betala tillbaka knappt 10 miljoner till de 85 investorerna på knappt två år.

Trots att Cinemaware idag är minst lika känt som US Gold och Electronic Arts, är det fortfarande en förhållandevis liten firma med totalt ca 75 anställda.

Men vilka anställda...

Cinemawares anställd har nästan alla en mer eller mindre brokig bakgrund (de är nämligen inte speciellt unga).

David Todd, Research and Development, har t ex designat och programmerat rymdfärjornas Satellite Rendezvous Navigation System och utfört lite jobb för IBM. Hans senaste uppgift var programmeringen av Lords of the Rising Sun...

Cinemaware är definitivt ingen tråkig arbetsplats. Men det har runnit mycket vatten under broarna sedan den trevande starten för drygt tre år sedan. Paret Phyllis och Bob Jacobs är visserligen fortfarande cheferna, han sköter om allt som har med produktutveckling att göra och hon tar hand om alla de "tråkiga rutinerna"...

"Men Bob och Phyllis startade egentligen med en idé om att bara speldesignen och artworken skulle skötas av Cinemaware själv", säger John Cutter från Product Development-avdelningen (har varit anställd sedan han designade Rocket Ranger).

"Det visade sig emellertid inte vara så lyckat, för det var nästan omöjligt att få frilansarna att producera det vi ville ha", fortsätter han. "Nu är det högst ett par av konverteringarna som ordnas externt".

Så arbetas det

Utvecklingsteamet för en typisk Cinemaware-titel består i regel av två programmerare, två grafiker, en "ljudman" samt speldesignern och producern. Det tar i genomsnitt ett år och upp till ett par hundra tusen dollar att utveckla ett program.

Men det är inte alltid det går lika smärtfritt. It Came From the Desert har Cinemaware redan lagt över två år på. Och på en direkt fråga om deras plan om en release 1989 fortfarande håller, svävar programmerarna betänkligt på målet...

Bob Jacob är Cinemawares kreativa hjärna, ideerna kommer på löpande band. "Så vi har en enorm lista med spelidéer liggande klara i arkivet", berättar Cutter.

När Cinemaware väl beslutat att föra en av Bobs idéer ut i livet, går uppgiften vidare till speldesignern (ofta John Cutter), som så tillsammans med en grafiker ska försöka göra idéerna mer handgripliga. Det kan ta flera månader att bli klar med en design.

Efter det att designen är klar är det artwork och ljud som gäller. I The Three Stooges använde Cinemaware t ex mer äm 1 Mb till ljudsidan (mer än 160 digitaliserade ljud). Det är vid denna del som producern kommer in i bilden.

Producern har det överordnade ansvaret för att tidsskemat håller och har alltid sista ordet i diskussionerna om det praktiska utförandet av spelet.

Produkterna från Cinemaware blir naturligtvis utvecklade på och till marknadens mest avancerade maskin, Amigan. Konverteringar till lillebror C64 och ett par andra maskiner sker först senare.

All grafik ritas på en Amiga med Deluxe Paint, oavsett vilket datorsystem den är avsedd för. Och programmerarna arbetar så vitt möjligt med Aztec C. Cinemaware använder sig inte av något skräddarsytt utvecklingssystem, då man menar att ett sådant allför snabbt utvecklar sig till att bli en sorts tvångströja. Ett litet system kallat Movie, som gör att grafikerna kan göra färdiga animationsavsnitt, utvecklades dock till Rocket Ranger. "Med Movie tar det högst en halvtimme från det programmeraren fått disketten i handen tills hela avsnittet är inlagt i spelet".

IT'S FILMTAJM, FOLKS!

Defender inte så bra

Det har med tiden kommit ett ansenligt antal Cinemaware-titlar på marknaden. "Phyllis och jag ska ju hålla svälten från dörren", ler Jacobs.

Hittills består en komplett Cinemawaresamling av följande titlar: Defender of the Crown, Sinbad and the Throne of the Falcon, S.D.I., King of Chicago, The Three Stooges, Rocket Ranger, TV Sports Football och Lords of the Rising Sun.

"Vi tycker själva att våra spel blir bättre och bättre för varje gång, men det är naturligtvis vissa som känns mycket bättre än andra. Min favorit hittills är Rocket Ranger, medan Defender of the Crown nog måste sägas utgöra lågvattenmärket. Defender var milt sagt inte direkt överbevisande, med tanke på att det faktiskt är en Amiga som står för datorkraften.

Det var nämligen så, att vi hade tvingats skriva under ett distributionskontrakt med Mindscape där det stod att det första spelet skulle levereras senast den 15 oktober 1986. Om fristen överskreds med bara en timme, skulle hela klabbet rasa ihop - vi skulle gå i konkurs.

Och så hände det som jag inte ens vågad drömma i mina vildaste mardrömmar. Vi förlorade två programmerare utan varsel. Där stod vi så den 1 juli,utan programmerare och med ett spel där bara designen och hälften av grafiken var klart.

Så jag ringde till en av mina 'gamla' kontakter, R.J. Michael, som hade varit med om att utveckla Intuition till Amiga. Jag erbjöd honom en stor säck pengar om han lovade att producera spelet på 6 veckor - och det gjorde han, gudskelov. Och detta är den enkla orsaken till att Amigaversionen är den svagaste.

Vill ni se den definitiva versionen, se på At... (namnet på en rivaliserande dator struket av hänsyn till läsare med svaga nerver).

Och vad angår det begränsade gameplayet, som många kritiserade, kan jag bara säga att våra spel designas så att de inte ska vara alltför omöjliga".

TV Sports

Men Cinemaware begränsar sig inte helt och hållet till interaktiva filmer. För ett tag sedan gav man sig på ytterligare en genre, sportspelen.

Deras första, TV Sports Football, tog också Amigamarknaden med storm. Som Bob Jacobs själv säger, gäller det "först och främst att få ut så mycket som möjligt av spelkonceptet. Spelidén måste fånga köparnas intresse, annars plockar de inte ens ner spelet från hyllan i affären för att titta på det."

Att Cinemaware alls gick in på sportspelsmarknaden berodde enligt Jacobs på arr John Cutter hade en massa nya idéer i skallen och ett par programmerare som var sportfanatiker. Största problemet, enligt

Det är billigare att prenumerera

Och enklare.

Cutter (som tidigare designat ett par sportspel för Gamestar), var att få Jacobs med på idén. För hur kan man intressera en filmfanatiker för sport?

"Jo, vi göt bara ett sportspel i stil med en TV-sändning, komplett med studiofolk, kommentator, presentation av spelarna och inte minst college- hejarklack (dansande collegetjejer)". Och då föll bossen för idén.

Men TV Sports Football är inte någon ensam svala. Två nya spel är på väg, och vi fick lov att ta en titt på nästa utgivning, TV Sports Basketball. Och precis som föregångaren är det hela nästan lika flott som i verkligheten. 5 spelare per lag på banan, flott animation och ljud, och programmerarna har inte glömt bort college-tjejerna...

Vi blev minst sagt imponerade.

Man lovar också att Basket inte blir det sista, nästa i serien blir TV Sports Baseball. Men Cutter var inte helt avvisande till idén om att göra en version av europeisk fotboll.

Ett par spionfoton av TV Sports Basketball.

Helst ska man vara långhårig om man jobbar på Cinemaware. Det här är en av grafikerna.

"Då kan jag ju få sitta och titta på video under arbetstiden igen...".

It Came From the Desert

Det andra spelet vi fick lov att ta en titt på, är ett spel helt i den klassiska Cinemawarestilen. It Came From the Desert utspelas i ett sömnigt californiskt ökensamhälle vid namn Lizard Breath. Mittpunkten i samhället är en stillastående stad totalt blottad på kulturell utveckling.

Den lugna tillvaron i öknen bryt emellertid abrupt då en meteor faller ner i lknen och det börjar strömma in rapporter om jätteinsekter.

Som "inflyttad" hjälte står du ensam i dina försök att övertyga invånarna om att de är i fara, och du måste lokalisera och besegra jätteinsekterna. Dramat blir heller inte mindre av att du utsätts för utpressare, kärlek och hemliga ökenkulter. Och spelaren får finna sig i att jobba sig igenom actionsekvenser med brutala knivslagsmål, höns på motorvägen, pyromaner och mycket annat.

Skulle du, mot förmodan, komma igenom It Came From the Desert med livet i behåll, ligger det redan en efterföljare och väntar på diskett. It Came From the Desert II innehåller ännu större doser dödliga insekter, hårdkokta skurkar och andra hemskiteter...

Spelet är absolut det närmaste Cinemaware hittills kommit till interaktiv film på dator. Det hela utspelas i real-time, på så sätt att när du väljer att gå till ett ställe är det något annat du missar. Så det gäller att tänka sig för - du har bara 15 dagar på dig innan staden blir uppäten. Spelet bjuder på 25 individuella roller, 38 platser (komplett med flygplats) och mer än 20 musikteman.

"Spelet är löst baserat på skräckfilmerna från 50-talet. Vi har behållit alla klichéerna från filmerna, som den sömniga förstaden, hungriga monster och hysteriska tonåringar som går direkt i armarna på monstren", säger teamet.

"Vi har hållit på mer än två år med spelet, och vi vågar lugnt påstå att det blir ett verkligt stort spel med massor med skräck och spänning. Optimalt sett borde du hoppa käpprakt upp ur stolen minst ett par gånger när du spelar. Men det har också blivit plats till en god portion humor och minst en lättklädd dam".

Och detta är inte det sista försöket i B-filmernas värld från Cinemaware. Nästa år ska det komma ett tvättäkta western-spel med massor med cowboys och indianer...

Mer än Amiga

"Jag vill göra interaktiv underhållning som är en hel upplevelse och inte bara ett spel, och det är Commodores Amiga den klart bästa maskinen till", säger Jacobs.

"Enda problemet med Amigan är att programmerarna har en förhållandevis brant inlärningskurva. Amigan är mer en förklädd minidator än en mikrodator, därför har de flesta programmerare stora problem med att utnyttja maskinen optimalt. Vi har jobbat med Amiga i tre år nu, men det är bara under det senaste året som vi blivit riktig kompis med maskinen.

Världen kommer under nästa år att få se Amigaprogram från Cinemaware som verkligen kommer att hamna på löpsedlarna. Vi har nu kommit till samma punkt om det tog 5 år för programmerarna att nå på Commodore 64. 1990 och 1991 kommer att bli de verkligt stora årgångarna för Amigasoftware, spår Jacobs.

Men efter det kommer det att bli svårt att hålla Bob kvar vid Amigan, för då kommer enligt Cinemaware allt att handla om CD-I och DVI. "Redan 1992 kommer 'äkta' interaktiv underhållning att slå ut allt annat. Och Cinemaware vill vara först och störst", säger Bob.

"Vi har engagerat David Riordan att leda vår Interactive Entertainment Group, och om det är någon som kan sätta fart under en ny teknologi så är det han", avslutar Bob Jacobs.

Vi fortsätter här vår serie om Amiga-BASIC och hur du får ut mest möjligt av den. Denna gång bjuder vi på hela tre programrutiner - och det är inte vilka som helst...

Av Flemming Steffensen

Om du är en av dem som gång på gång fått höra en massa kritik om BASIC-programmering, har du säkert ibland undrat om det överhuvudtaget *går* att göra "riktiga" program i BASIC.

Vi kanske med en gång ska slå fast, att några BASIC- versioner av OutRun, Datastorm eller Xenon är det knappast lönt att ge sig på. BASIC är helt enkelt för långsamt för sådana spel. Men program som Defender of the Crown, Ports of Call och Football Manager kunde faktiskt ha skrivits i BASIC.

Det avgörande för resultatet är att du utnyttjar BASIC fullt och helt. Det är de smarta rutinerna som fäller utslaget! Det är här COMputer (än en gång) ilar till din hjälp. Vi fortsätter med att plocka upp kaniner ur cylinderhatten i form av programrutiner, som tillsammans med dem vi tidigare gett hjälper dig en bra bit på väg mot de perfekta BASIC-programmen.

Pilen i centrum

Program 1 visar hur du lätt och elegant gör din egen design av Workbench-pilen. Det är väl inget nytt kanske du tänker, det kan man ju göra i preferences. Jo, visst kan du det, men hur skulle det vara att kunna ge pilen flera utseenden i samma program?! Det är det vi kan bjuda på här!

När du kör programmet öppnas tre fönster. Beroende på vilket du klickar i kommer pilen att ändra form. Det är viktigt att du förstår att pilens form är knuten till det aktuella fönstret. Även om du flyttar pilen in i ett annat fönster är det först när du klickar som pilen ändrar form.

Men hur räknas datastrukturen ut? Jo, först kommer två nollor, sedan pilens data och till sist två nollor till. Nollorna ger knappast anledning till problem (vilken tillit!), men hur är det med resten? Det hela bygger på det faktum att pilen är en sprite som består av två plan, vilket ger fyra färger. Den första färgen är alltid genomskinlig, de tre återstående är paletterna 17, 18 och 19. Dessa tre färger kan du alltså ändra med PALETTE-kommandot. VArje datarad innehåller två värden för plan 1 respektive plan 2. Hur pilens data räknas ut närmare återkommer vi till i en senare artikel.

Systemrutinen

Själva bytet av pilens o‹form sker via kalla av en systemrutin, SetPointer(). Innan vi kan använda rutinen ska vi göra plats till våra data, och det gör vi med AllocRaster (bredd, storlek). Bredd är 16 eller under och storlek är antalet data (inkl nollor). Så läser och pokar vi in våra data och kallar:
SetPointer (fönster, datastruktur, höjd, bredd, x, y)

Höjd och bredd är pilens storlek i pixels medan x och y är offset-värden för den s k Hot Spot. Du känner ju till att det inte är hela pilen som räknas när du klickar på någor, utan bara en punkt. Det är denna punkt som ytterst passande kallas Hot Spot. Och placeringen av denna punkt kan du alltså bestämma med x och y. Värdena mäts från standardplaceringen - översta vänstra hörnet (0,0). Ett negativt värde på x flyttar punkten åt höger och ett negativt värde på y flyttar punkten nedåt.

Givetvis ska det också vara möjligt att återställa pilens ursprungliga form. Detta görs med ClearPointer(). Kom också alltid ihåg att frige det reserverade minnet med FreeRaster(). Lägg märke till att du också kan få en osynlig pil om du låter alla data vara nollor.

Ned med skärmen

Program 2 kan gott gå och bli din favoritrutin så småningom. Programmet visar i all sin enkelhet bara hur du kan scrolla skärmen upp och ned. Lägg märke till hur glidande BASIC klarar uppgiften, och som du kan se kan det också gå riktigt snabbt - även i BASIC!

Även detta program kallar en systemrutin, i det här fallet MoveScreen(). Först beräknar vi skärmens adress genom adress=PEEKL (WINDOW(7)+46). Så är vi klara att kalla rutinen:
MoveScreen (adress,x,y)

Värdet x ignoreras och ska vara 0, medan vädet y flyttar skärmen ned eller upp.

Program 2 öppnar två skärmar, en med svart bakgrundsfärg och en med röd. Den röda skärmen scrollas ned och upp med varierande hastighet medan det ritas linjer på den röda och boxar på den svarta.

Russinet i kakan

Till sist, men absolut inte minst, ska vi leka med Coppern. Det skulle tyvärr ta upp alltför mycket plats om vi punkt för punkt skulle gå igenom hur program 3 fungerar, det är en ganska komplicerad programmering som för att förstås kräver en hel del förkunskap om Commerns arbetsmetoder. Men förtvivla icke - programmet är enkelt att använda.

När du kör programmet öppnas en skärm och ett litet fönster. Efter en liten stund händer något som du nog inte trodde var möjligt i BASIC! Hela skärmen (förutom fönstret) fylls med färgade linjer. Du kan skriva och rita som vanligt i fönstret, det är nämligen bakgrundsfärgen som skiftas ned längs skärmen. Det betyder också att du bevarar linjerna i bakgrunden vid CLS-kommando.

Detta kan ge en hel del otroligt flotta effekter och passar fint till program som du vill ge ett proffsigt utseende. Och som vanligt är det COMputer som ger dig det du inte trodde var möjligt!

FÅ KOLL PÅ BASIC!

```
'*****************************************
'***     Program 1:  Dina Egna Pekare    ***
'***                                     ***
'*****************************************

'NB: graphics.bmap och intuition.bmap ska ligga i
'    LIBS eller den aktuella directoryn

LIBRARY "graphics.library"
LIBRARY "intuition.library"

DEFLNG a-z
DECLARE FUNCTION AllocRaster() LIBRARY

pointer=AllocRaster(16,34)
IF pointer=0 THEN Fel
pointer2=AllocRaster(16,160)
IF pointer2=0 THEN Fel

ON BREAK GOSUB Ut: BREAK ON
SCREEN 2,640,256,3,2
WINDOW 2,"Window 1",(0,0)-(170,75),15,2
w2=WINDOW(7)
WINDOW 3,"Window 2",(110,100)-(400,200),15,2
w3=WINDOW(7)
WINDOW 4,"Window 3",(320,20)-(570,110),15,2
PRINT "(Ctrl) (C) to stop"

PALETTE 17,0,1,0:PALETTE 18,1,0,1
PALETTE 19,0,1,1

FOR d=0 TO 64 STEP 4
  READ a,b
  POKEW pointer+d,a:POKEW pointer+d+2,b
NEXT

a=&HF00F:b=&HFFFF
FOR d=0 TO 320 STEP 4
  POKEW pointer2+d,a:POKEW pointer2+d+2,b
NEXT

CALL SetPointer(w2,pointer,15,15,-15,-15)
CALL SetPointer(w3,pointer2,80,15,-7,-75)

Loop:
GOTO Loop

Fel:
BEEP:PRINT "Not Enough RAM"

Ut:
CALL ClearPointer(w3):CALL ClearPointer(w2)
CALL FreeRaster(pointer,16,34)
CALL FreeRaster(pointer2,16,160)
WINDOW CLOSE 2:WINDOW CLOSE 3:WINDOW CLOSE 4
SCREEN CLOSE 2
LIBRARY CLOSE
LIST
END


'Data for Pointer

DATA 0,0
DATA &H0000,&H0000
DATA &H0FF8,&H0000
DATA &H1004,&H0FF8
DATA &H2FFA,&H1FFC
DATA &H5004,&H3006
DATA &HA002,&H6002
DATA &HA000,&H6000
DATA &HA000,&H6000
DATA &HA000,&H6000
DATA &HA006,&H6006
DATA &H5008,&H300A
DATA &H2812,&H1814
DATA &H17E4,&H0FE8
DATA &H0808,&H07F0
DATA &H07F1,&H0001
DATA 0,0


'*********************************
'***  Program 2: Move Screen   ***
'*********************************

'NB: intuition.bmap ska ligga i
'    LIBS eller den aktuella directoryn

DEFLNG a-z
ON BREAK GOSUB slut:BREAK ON
LIBRARY "intuition.library"
SCREEN 1,320,200,5,1
WINDOW 3,,,0,1
PALETTE 0,0,0,0:PALETTE 1,0,0,0
SCREEN 2,320,200,5,1
WINDOW 2,,,0,2
PALETTE 0,1,0,0:PALETTE 1,1,0,0:COLOR 3
PRINT "(Ctrl) (C) to stop..."
t#=TIMER: WHILE TIMER(t#+3:WEND
CLS

adress=PEEKL(WINDOW(7)+46)

s=1:d=1

loop:
FOR w=1 TO 256/s
  CALL MoveScreen(adress,0,s)
NEXT
LINE (RND*320,RND*200)-(RND*320,RND*200),RND*30
FOR w=1 TO 256/s
  CALL MoveScreen(adress,0,-s)
NEXT
WINDOW OUTPUT 3
LINE (RND*320,RND*200)-STEP(RND*50,RND*50),RND*30,bf
WINDOW OUTPUT 2
s=s+d : IF s>50 OR s<2 THEN d=-d
GOTO loop

slut:
WINDOW CLOSE 2:WINDOW CLOSE 3
SCREEN CLOSE 2:SCREEN CLOSE 1
LIBRARY CLOSE
PRINT "slut..."
END

p'*************************************
'***   Program 3: Copper from Basic  ***
'***                                 ***
'*************************************

'NB: exec.bmap,graphics.bmap och intuition.bmap
'    ska ligga i LIBS eller den aktuella
'    directoryn

LIBRARY "exec.library"
LIBRARY "graphics.library"
LIBRARY "intuition.library"

DECLARE FUNCTION ViewPortAddress& LIBRARY
DECLARE FUNCTION AllocMem& LIBRARY

SCREEN 2,640,256,3,2
WINDOW 2,,(100,100)-(500,200),0,2
PRINT "Wait..."

x%=20:ym%=400:c%=384

Cop&=AllocMem&(12,65539&)
IF Cop&=0 THEN Fel

FOR w%=5 TO ym%-1
  CALL CWait(Cop&,w%,x%)
  CALL CBump(Cop&)
  ' Change 36 i next line to
  ' f.ex 2,4,8,9,12,24,64 etc.
  CALL CMove(Cop&,c%,w%*36)
  CALL CBump(Cop&)
NEXT

CALL CWait(Cop&,10000,256)
vp&=ViewPortAddress&(WINDOW(7))
POKEL vp&+20,Cop&
CALL RethinkDisplay
PALETTE 1,0,0,0
CLS:PRINT "Look at this line"
t#=TIMER: WHILE TIMER(t#+5:WEND
CLS
PRINT "Colors in background stays with CLS"
t#=TIMER: WHILE TIMER(t#+5:WEND
CLS:PRINT "Click here to end"

WHILE MOUSE(0)=0:WEND

Ut:
WINDOW CLOSE 2:SCREEN CLOSE 2:LIBRARY CLOSE
LIST
END

Fel:
PRINT "NO RAM!!!"
GOTO Ut
```

CLI SKOLAN 3

Och här är den! Som vi lovade i förra avsnittet kommer vi här med ett BASIC-program som en gång för alla löser alla dina problem med att göra CLI-diskar.

Av Flemming Steffensen

```
'*******************************************************
'*            COMputers  CLI-disk Maker                *
'*             av  Flemming Steffensen                 *
'*                   September 1989                    *
'*  OBS: Detta program ska startas från CLI!!!!        *
'*******************************************************



SCREEN 1,640,250,1,2
WINDOW 1,"COMputers CLI-disk Maker",,0,1
PALETTE 0,0,0,0:PALETTE 1,.5,.5,.5
ON BREAK GOSUB out:BREAK ON
DIM SHARED var(20)
LINE(0,200)-(640,200),1 : LINE(0,202)-(640,202),1

MAIN:
a=0:ERASE var:DIM SHARED var(20)
RESTORE Text:GOSUB PrintLoop
IF var(11) THEN GOTO MAIN
RESTORE Text2:GOSUB PrintLoop
GOSUB InitLib
c$="copy c:run ram:":GOSUB ExCom
c$="assign c: ram:":GOSUB ExCom
m$="Stick i din kopia av Workbench i drive DF0:"
LOCATE 12,(80-LEN(m$))/2:PRINT m$;
m$="Tryck en tangent"
LOCATE 28,(80-LEN(m$))/2:PRINT m$;
GetAny
LINE(1,80)-(638,198),0,bf
LINE(2,203)-(638,248),0,bf
c$="DF0:C/assign C: DF0:C":GOSUB ExCom
IF var(4)=0 THEN font
NAME "df0:utilities/Notepad" AS "df0:c/Notepad"

GOTO speak
font:
IF var(5)=0 THEN
  c$="delete df0:fonts all":GOSUB ExCom
  KILL"df0:libs/diskfont.library"
END IF
speak:
IF var(9)=1 THEN NAME "df0:utilities/say" AS "C:say"
IF var(9)=0 THEN
    KILL "df0:L/Speak-Handler"
    KILL "df0:devs/narrator.device"
    KILL "df0:Libs/translator.library"
END IF
NAME "df0:utilities/More" AS "C:More"
c$="delete df0:utilities all":GOSUB ExCom
c$="delete df0:trashcan all":GOSUB ExCom
NAME "df0:Prefs/Preferences" AS "C:Preferences"
c$="delete df0:Prefs all":GOSUB ExCom
NAME "df0:System/Format" AS "C:Format"
IF var(1)=1 THEN
  NAME "df0:System/NoFastMem" AS "C:NoFastMem"
END IF
NAME "df0:System/SetMap" AS "C:SetMap"
NAME "df0:System/DiskCopy" AS "C:DiskCopy"
c$="delete df0:System all":GOSUB ExCom
KILL "df0:L/Aux-Handler"
KILL "df0:L/FastFileSystem"
KILL "df0:L/Pipe-Handler"


IF var(8)=0 THEN
    KILL "df0:L/Newcon-Handler"
    KILL "df0:L/Shell-Seg"
    KILL "df0:C/NewShell"
END IF
KILL "df0:Shell"
IF var(3)=0 THEN
    KILL "df0:devs/printer.device"
    c$="delete df0:devs/printers all"
    GOSUB ExCom
END IF
IF var(6)+var(7)=0 THEN Startup
RESTORE RAMdata
OPEN "df0:RAMDISK.BATCH" FOR OUTPUT AS 1
n$="":t$="path RAM:c add":f=0
Makeit:
WHILE n$<>t$
  READ n$
  PRINT #1,n$
WEND
IF var(6)=1 AND f=0 THEN
  n$="":t$="copy c:run ram:c":f=1
  GOTO Makeit
END IF
CLOSE 1
Startup:
c$="delete df0:s/#?":GOSUB ExCom
OPEN "df0:s/Startup-Sequence" FOR OUTPUT AS 1
PRINT#1,"Addbuffers df0: 10"
PRINT#1,"FF >NIL: -0"
PRINT#1,"makedir ram:t"
PRINT#1,"assign t: ram:t"
IF var(2)=1 THEN
  PRINT#1,"setclock opt load"
END IF
IF var(10)=1 THEN
  PRINT#1,"Setmap s"
END IF

IF var(6)+var(7)>0 THEN
  PRINT#1,"Execute SYS:RAMDISK.BATCH"
END IF
IF var(8)=1 THEN
  PRINT#1,"resident >NIL: CLI L:Shell-Seg SYSTEM pure add"
  PRINT#1,"mount NEWCON:"
  PRINT#1,"NewShell NewCon:0/0/640/256/CLI-disk"
END IF
CLOSE 1
IF var(10)=1 THEN
  c$="copy df0:c/copy RAM:":GOSUB ExCom
  c$="assign C: ram:":GOSUB ExCom
```

Som du kan se är det ett långt (men minst lika bra) program. Det betyder att vi inte har något 'Månadens CLI-kommando' denna gång och att vi här i texten inskränker oss till en vägledning i hur du använder programmet.

Och det är lika bra att säga det med en gång: Kolla din inskrivning av programmet många gånger innan du kör det. De är nästan omöjligt att inte göra några fel i en så stor listning. Var försiktig!

Lägg också märke till att programmet kräver Workbench 1.3. Har du inte denna är det kanske dags att få köpt den nu - förbättringarna från version 1.2 är många. Du behöver INTE Kickstart 1.3 för att köra programmet.

Programmet

Programmet tillverkar dina CLI-diskar åt dig - helt efter ditt eget önskemål. Givetvis kan man inte ta upp alla möjligheter, men de allra flesta torde kunna använda de CLI-diskar som programmet gör utan att någonsin behöva göra ändringar på dem.

Programmet kan hålla reda på extraminne, batteribackup, printerdrivers, Notepad, fonts, 2 RAM-diskar, Shell, talsyntes och tangentbordsdriver. Med andra ord: ALLT!

Tillvägagångssätt

Och nu till tillvägagångssättet. För att vara säker på att alla kan följa med, tar vi det ända från början. Vi utgår från att du bara her en diskettstation (har du fler kan det vara en god idé att slå av den/dem för att spara minne - speciellt om du bara har 512 kb RAM).

Först startar du Amigan med Workbench 1.3. Men innan vi läser in Amiga BASIC måste vi ta hand om ett litet problem. Det är ju så, att Amiga BASIC befinner sig på Extras-disketten (den som följer med Workbench) tillsammans med en massa andra filer. Disketten är nästan full, så det finns inte plats för vårt program på den.

Det du ska göra, är att ta en kopia av Extras-disketten och radera alla programmen i den directory (låda) som heter BasicDemos. Om du inte vet hur man ska göra, är receptet som följer:

1. Sätt i Extras-disketten i diskettstationen
2. Klicka en gång (och bara en gång) på diskettikonen
3. Välj Duplicate från Workbenchmenyn längst till vänster
4. Följ instruktionerna

När disketten är kopierad kan du se att den nya disketten heter "copy av Extras 1.3". Klicka två gånger på denna ikon, varpå diskettens innehåll visar sig. Klicka två gånger på BasicDemos-lådan - innehållet kommer fram på skärmen. Håll ner SHIFT-knappen (hela tiden) och klicka på alla ikonerna i BasicDemos-fönstret - de blir svarta. Släpp SHIFT och välj Discard från Workbench-menyn, så raderas alla filerna. När detta är klart kan du sätta i Workbench-disketten igen.

När vi nu ändå är igång kan vi lika gärna fortsätta med att kopiera Workbench-disketten (det är denna kopia som så småningom ska bli vår CLI-disk). Kopiera Workbench-disketten genom att upprepa punkt 2-4 som ovan.

Nu har du de disketter som ska användas. Det är original Workbench-disketten, en kopia av denna och en kopia av Extras-disketten med plats för programmet. Då kan vi börja.

Sätt i kopian av Extras-disketten, öppna disketten och läs in AmigaBASIC genom att klicka två gånger på ikonen. Knappa så in programmet - du får inte köra programmet än, för då går datorn ned! Göm programmet under titeln CLI-PROGRAM. Kolla din inskrivning grundligt.

Det stora ögonblicket

Om du istället för ovanstående sätt valde att kopiera Workbench-disketten på något annat sätt, ska du veta att programmet inte fungerar om kopian har samma namn som originaldisketten. Den SKA heta något annat (t ex "copy of Workbench 1.3").

En annan viktig sak, är att programmet inte fungerar om du läser in AmigaBASIC genom att klicka på ikonen. För att köra programmet MÅSTE du starta AmigaBASIC från CLI.

Gör såhär:

1. Stäng av Amigan och sätt i original-Workbenchen (inte kopian)

```
    RESTORE Text3
    GOSUB PrintLoop
    c$="copy df0:devs/keymaps/s ram:":GOSUB ExCom
    RESTORE Text4
    GOSUB PrintLoop
    c$="copy ram:s df0:devs/keymaps/s":GOSUB ExCom
    KILL "RAM:s":KILL "RAM:copy"
    c$="df0:c/assign C: df0:C":GOSUB ExCom
  END IF
  Printerval:

  IF var(3)=1 THEN
    c$="copy df0:c/copy RAM:":GOSUB ExCom
    c$="assign C: ram:":GOSUB ExCom
    c$="copy df0:c/list ram:":GOSUB ExCom
    RESTORE Text3
    GOSUB PrintLoop
    DIM p$(40)
    c$="list >ram:printers df0:devs/printers quick"
    GOSUB ExCom
    OPEN "ram:printers" FOR INPUT AS 1
    t=0
    WHILE NOT EOF(1)
      LINE INPUT #1,p$(t)
      t=t+1
    WEND
    CLOSE 1
    t=t-2 :a=1:b=2
    FOR w=1 TO t
      LOCATE 2+a,b:PRINT USING "##. &";w,p$(w)
      IF b=2 THEN b=40 ELSE b=2:a=a+1
    NEXT
  p:
    COLOR 1,0
    m$="Ange printer (1 -"+STR$(t)+") "
    LOCATE 28,(80-LEN(m$))/2:PRINT m$;
    INPUT k$:IF k$="" THEN p
    v=VAL(k$):IF v<1 OR v>t THEN p
    KILL "RAM:printers"
    s$="df0:devs/printers/"+p$(v):t$="RAM:"+p$(v)
    c$="copy "+s$+" "+t$:GOSUB ExCom:SWAP s$,t$
    LINE(1,1)-(638,198),0,bf:LINE(2,203)-(638,248),0,bf
    RESTORE Text4
    GOSUB PrintLoop
    c$="copy "+s$+" "+t$:GOSUB ExCom
    ERASE p$
    RESTORE Text5
    GOSUB PrintLoop
    c$="df0:c/assign C: df0:c":GOSUB ExCom
  END IF

  SLUT:
  m$="Wait..."
  LOCATE 28,(80-LEN(m$))/2:PRINT m$;
  c$="relabel df0: name "+CHR$(34)+"CLI-disk"+CHR$(34)
  GOSUB ExCom
  RESTORE Slutkom
  s:
  READ c$:IF c$="x" THEN s2
  GOSUB ExCom:GOTO s

  s2:
  t#=TIMER:WHILE TIMER<t#+10:WEND
  RESTORE Text6
  GOSUB PrintLoop
  out:
  CLS:END

  PrintLoop:
  READ y,m$:IF y=-1 THEN RETURN
  IF LEFT$(m$,1)="#" THEN
    flag$=MID$(m$,2,1):m$=RIGHT$(m$,LEN(m$)-2)

    IF flag$="k" THEN CALL GetAny
    IF flag$="j" THEN CALL GetJN
    IF flag$="s" THEN CALL StoreV
    IF flag$="c" THEN LINE(1,1)-(638,198),0,bf
    IF flag$="f" THEN LINE(2,203)-(638,248),0,bf
  END IF
  IF LEN(m$)>0 THEN LOCATE y,(80-LEN(m$))/2:PRINT m$;
  GOTO PrintLoop

  InitLib:
  f$="Execute"+CHR$(0)
  f$=f$+CHR$(255)+CHR$(34)+CHR$(2)
  f$=f$+CHR$(3)+CHR$(4)+CHR$(0)
  CHDIR "RAM:"
  OPEN "dos.bmap" FOR OUTPUT AS 1
  PRINT #1,f$;
  CLOSE 1
  LIBRARY "dos.library"
  CHDIR "DF0:"
  RETURN

  ExCom:
  CALL Execute&(SADD(c$+CHR$(0)),0,0)
  RETURN

  SUB GetAny STATIC
  ClearBuffer
  WHILE INKEY$="":WEND
  END SUB

  SUB ClearBuffer STATIC
  WHILE INKEY$<>"":WEND
  END SUB

  SUB GetJN STATIC
  SHARED k$
  k$=""
  ClearBuffer:x=CSRLIN:y=POS(d)
  g:
    LOCATE x,y+1
    k$=UCASE$(INKEY$):IF k$="" THEN g
    PRINT k$
    IF k$<>"J" AND k$<>"N" THEN g
  END SUB
```

CLI SKOLAN 3

2. Starta Amigan
3. Så snart den blå skärmen visar sig (diskettstationen jobbar fortfarande) håller du nere CTRL och D samtidigt, så kommer du ut i CLI
4. Sätt in din kopia av Extras (den med programmet) i driven
5. Skriv df0:AmigaBasic df0:CLI-PROGRAM
6. Tryck RETURN (Enter)

Så läses AmigaBASIC in och programmet startas automatiskt.

Programmet ber dig först att sätta i original-Workbenchen (den du startade Amigan med). Du ska skifta diskett flera gånger medan programmet kör, så kom ihåg att ALLTID vänta tills dioden slocknat innan du plockar ur en diskett. Likaså ska du vänta tills dioden slocknat när du lagt i en ny diskett, innan du trycker på en tangent. Får du en blå skärm (standard Workbench-färg) kan du försöka med att hålla ner VÄNSTER Amigatangent och M, så är problemet måhända löst.

Frågor och åter frågor

Nu ställer programmet en massa frågor till dig. Det är här du bestämmer hur din kommande CLI-disk ska se ut. Om du inte har någon RAM-expansion ska du försöka svara nej på så många frågor som möjligt. Speciellt viktigt är att du svarar nej på frågan om stor RAM-disk, i annat fall får du allvarliga problem med minnet.

Frågan om du har en batteri-backup används för att avgöra om CLI-disken ska läsa in datum och tid automatiskt.

Senare frågas om du vill använda SHELL. Om du inte vet vad det är, kan vi kort säga att det är en mer användarvänlig variant av CLI (vi går igenom Shell i en senare artikel). Svarar du ja, kommer Shell automatiskt att läsas in när du använder din CLI-disk.

Så småningom kanske programmet ber dig om att sätta i Extras-disketten. Här spelar det ingen roll om det är kopian med programmet eller originalet du stoppar i.

Har du svarat ja på frågan om printer, kommer så småningom en lista över de skrivare du kan välja mellan. Finns inte din egen skrivare med kan du försöka med EpsonOld, den fungerar med många skrivare.

När programmet är klart kommer du ut i AmigaBASIC och din CLI-disk är redo att användas.

Jungfruturen

Nu kan du som sagt använda din CLI-disk. Stäng av Amigan och slå på den igen eller gör en reset genom att hålla nere CTRL och de två Amigatangenterna samtidigt. Din CLI-disk ska nu starta automatiskt. När allt väl är inläst kan du börja arbeta.

Har du valt att använda printer ska du komma ihåg att starta preferences (skriv Preferences och tryck Return). Välj Change Printer och välj den skrivare du ska använda. Välj OK och SAVE, och så är du klar för utskrift.

Har du valt en liten eller stor RAM-disk har det kopierats upp några kommandon i RAM:. Det betyder att du nu kan skifta diskett och ändå använda kommandon som TYPE och DIR. För att se vilka kommandon du har till rådighet kan du skriva DIR RAM:c och trycka Return.

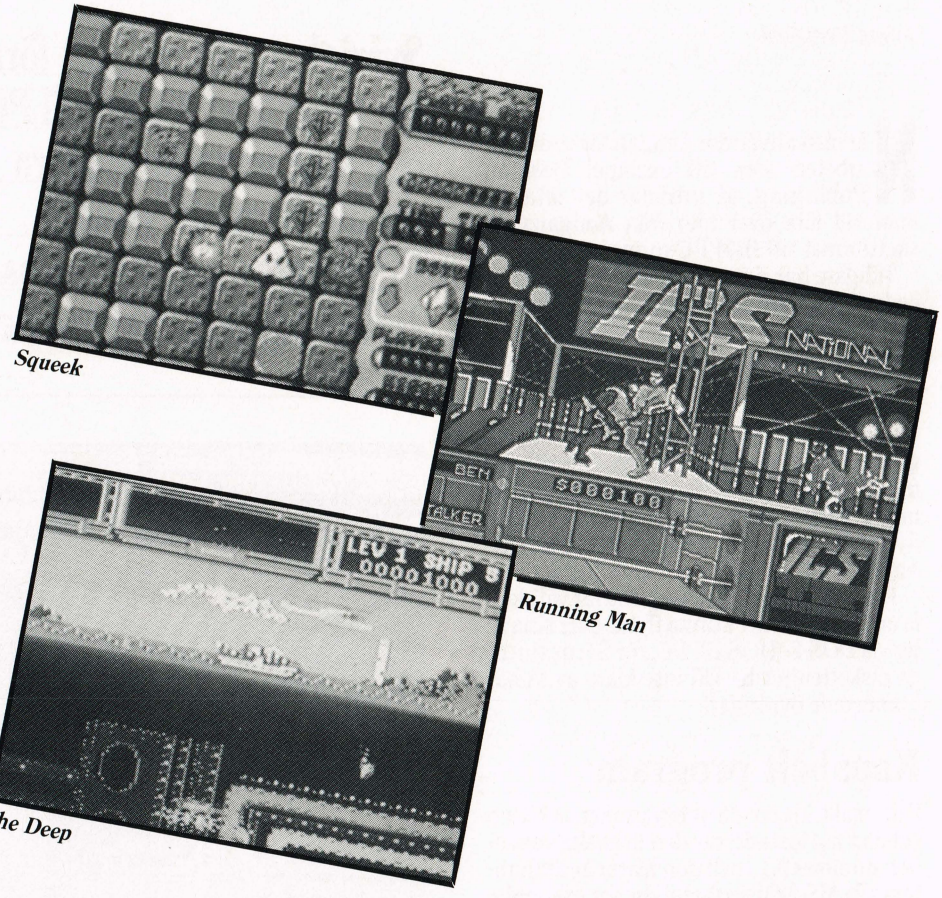
I kommande artiklar ser vi närmare på vad du kan använda denna din CLI-disk till och hur du gör dina egna BATCH-filer. BATCH-filer kan lösa många uppgifter som annars kräver mycket inknappning.

Mycket nöje och lycka till! ■

```
SUB StoreV STATIC
SHARED k$,a
a=a+1
IF k$="J" THEN var(a)=1 ELSE var(a)=0
END SUB


Text:
DATA 2,"#c",2,"#f",2,"CLI-disk Maker"
DATA 3,"Av Flemming Steffensen"
DATA 6,"Har du din original Workbench-diskett"
DATA 8,"i drive DF0: ? Annars ska du i med den nu!!"
DATA 28,"Tryck en tangent"
DATA 28,"#k",28,"#f",28,"#c"
DATA 2,"Du ska nu starta tillverkningen av din egen CLI-disk."
DATA 4,"Var beredd på att du ska byta  "


DATA 6,"diskette ibland - Du ska ha  3 disketter:"
DATA 8,"1. Din original Workbench-diskett (skrivskyddad)"
DATA 9,"2. Din Workbench Extras diskett (den med BASIC)  "
DATA 10,"3. Din kopia av Workbench-disketten, som:              "
DATA 12,"inte får vara skrivskyddad!!"
DATA 28,"Tryck en tangent  (CTRL C = Avbryt)"
DATA 28,"#k",28,"#f",28,"#c"
DATA 2,"OK, let`s go! Som start ska du svara på"
DATA 4,"några generella frågor (giltiga svar = 'j' - "
DATA 6,"JA och 'n' - NEJ).              Vars'go...          "
DATA 28,"Svara J eller N"
DATA 9,"1.  Har du extra 512K RAM?"
DATA 9 ,"#j",9,"#s"
DATA 11,"2.  Har du batteri-backup ?"
DATA 12,"(normalt JA, om du har > 512K)"
DATA 11,"#j",11,"#s"
DATA 13,"3.  Vill du kunna ha en printer?"
DATA 13,"#j",13,"#s"
DATA 15,"4.  Vill du ha NotePad?"
DATA 15,"#j",15,"#s"
DATA 17,"5.  Vill du ha Fonts? "
DATA 17,"#j",17,"#s"
DATA 19,"6.  Vill du ha en stor RAM-disk?"
DATA 19,"#j",19,"#s"
DATA 21,"7.  Vill du ha en liten RAM-disk?"
DATA 21,"#j",23,"#s"
DATA 4,"#c"
DATA 4,"8.  Vill du ha SHELL?"
DATA 4,"#j",4,"#s"
DATA 6,"9.  Vill du ha talsyntes?"
DATA 6,"#j",6,"#s"
DATA 8,"10. Vill du starta med svenska tecken?"
DATA 8,"#j",8,"#s"
DATA 12,"11.  Vill du gå igenom valproceduren igen?"
DATA 12,"#j",12,"#s"
DATA -1,x
Text2:
DATA 2,"#c",2,"#f",2,"Så, nu jobbar jag...."
DATA 6,"WAIT...."
DATA -1,x
Text3:
DATA 2,"#c",2,"#f",2
DATA "In med Workbench Extras disketten i drive DF0:"
DATA 28,"Tryck en tangent",28,"#k",28,"#f",28,"#c"
DATA -1,x
Text4:
DATA 2,"In med din kopia av Workbench-disketten i drive DF0:"
DATA 28,"Tryck en tangent",28,"#k",28,"#f",28,"#c"
DATA -1,x
Text5:
DATA 12,"REMEMBER:"
DATA 14,"Vid 1:a  bootningen (uppstart från)"
DATA 16,"din nya CLI-disk, ska du skriva PREFERENCES"
DATA 18,"och ta fram den printer du vill ha.          "
DATA 28,"Tryck en tangent",28,"#k",28,"#f",28,"#c"
DATA -1,x
Text6:
DATA 28,"#f",28,"#c"
DATA 2,"Okay, din CLI-disk står klar "
DATA 4,"Hoppas resultatet blivit okej, annars får du"
DATA 6,"starta igen från start. "
DATA 28,"Tryck en tangent",28,"#k",28,"#f",28,"#c"
DATA -1,x
RAMdata:
DATA "makedir RAM:c
DATA "copy c:copy ram:c"
DATA "copy c:dir ram:c"
DATA "copy c:delete ram:c"
DATA "copy c:type ram:c"
DATA "path RAM:c add"
DATA "copy c:info ram:c"
DATA "copy c:echo ram:c"
DATA "copy c:list ram:c"
DATA "copy c:rename ram:c"
DATA "copy c:makedir ram:c"
DATA "copy c:run ram:c"
Slutkom:
DATA "delete ram:#?"
DATA "delete df0:c/#?.info"
DATA "delete df0:s/#?.info"
DATA "delete df0:t"
DATA "delete df0:Empty"
DATA "delete df0:Expansion"
DATA "delete df0:Empty.info"
DATA "delete df0:Prefs.info"
DATA "delete df0:Trashcan.info"
DATA "delete df0:Expansion.info"
DATA "delete df0:Utilities.info"
DATA "delete df0:System.info"
DATA "delete df0:Ramdisk.batch.info"
DATA "x"
```

Snabbisar

Snabbisar är vår nya sektion för spel som inte kommer med i våra normala spelsektioner. Korta recensioner av spel som kommit in sent, är konverteringar av spel som vi redan testat i ett annat format eller helt enkelt inte håller för en lång speltest.



Squeek

Running Man

The Deep

Running Man

Jo, Schwarzenegger är på gång igen i ett nytt spel. Den här gången heter spelet Running Man, och om inte annat är det i varje fall annorlunda i vissa aspekter.

Inledningen är en veritabel orgie i grafik och ljud som varar länge, länge, länge... Digitalisering är ordet, och det hela är verkligt flott. Men kommer vi till själva spelet är det inte så mycket att hurra för.

Spelet går ut på att du ska springa och hoppa på diverse plattformar medan du sparkar och slår omkring dig. Problemet är bara att du måste hoppa så exakt så det närmast är löjligt, du har bara ett par pixel på dig att utföra din rörelse. Till och med värste actionfreaken på redaktionen, en som kan sitta hela dygn med Datastorm eller Blood Money, tröttnade efter fem minuter.

Sammanfattningsvis: Grafik - jodå, ljud - visst, action - njaää, fängslande - nej. Spelgrafiken är flott och annorlunda, synd bara att de slösat bort den på en så taskig handling. Grafik och ljud enbart hade gett 40% mer i betyg. Totalbedömning: 35%

Squeek

Fransmännen kan verkligen göra spel som skiljer sig från dem vi vant oss vid från England eller USA. Och det utfaller oftast till fransosernas fördel...

Squeek är en söt blandning mellan Pacman, Penguin och Painter, och resultatet har blivit riktigt kul. Låter man intron köra presenterar Squeekfiguren alla monster och bonusar man kan få tag på, och det tar tid. Här finns så mycket olika saker att man knappt har en chans att hålla reda på ens en bråkdel, så man plockar allt i hopp om det bästa.

Snygg grafik, söt melodi som snabbt blir läskigt irriterande, roliga spelsekvenser, tjusig scrollning - spelet har det mesta. Men när man väl slutligen fått game over måste man tröska sig igenom alla nivåerna igen från början - och det finns 99 stycken... Tedious, brukar det heta i de engelska recensionerna. Ett bagatellspel som man gärna slösar bort många timmar i sällskap med.

Totalbedömning: 78%

The Deep

- har vi tidigare testat i C64-versionen, och allt som gällde då gäller i stort sett fortfarande. För er som inte läst C64-testen kan vi avslöja, att spelet är en sorts modern variant på det gamla videospelet där det gällde att droppa sjunkbomber på ubåtar.

Jovisst, Amigaversionen är grafiskt sett betydligt snyggare, men inte ens en snygg grafik kan hjälpa upp en hopplöst föråldrad handling. Spelet är rent ut sagt dött som en sill. Det finns visserligen en hel del action i spelet, men man har helt enkelt ingen lust att spela det! Det finns alldeles för många bra andra spel som man mycket hellre vill slösa sin dyrbara tid på.

Totalbedömning: 12%

Forts sid 41

Gränsbrytare av format

Jonny Bergdahl

Svårigheterna med att föra över filer från Amiga till PC och vice versa är numera ett minne blott, enligt tillverkaren av Dos-2-Dos. Håller programmet vad det lovar?

När man använder Amigan för seriösare arbeten som till exempel Desktop Publishing, så inträffar det ofta att man vill föra över filer från Amigans diskettformat till IBM PC-format.

Tidigare har detta varit en långsam procedur i och med att filerna har fått skickas via långsamma modem. Central Coast Software har dock haft lösningen på detta problem ett stycke tid nu, med programmet DOS-2-DOS.

DOS-2-DOS utläses dos-to-dos, klarar galant av denna uppgift. Flytta filer till och från IBM:s 3,5 och 5,25- tumsformat går utmärkt. Dessutom klarar programmet av att flytta filer mellan Ataris diskettformat och Amigan, vad detta nu ska vara bra för?

Fast det är klart, Atari-ägarna får väl skaffa en Amiga för att kunna flytta över sina filer via DOS-2- DOS till PC, för ST:ns fjuttiga diskettenhet lär väl inte klara av så här avancerade övningar!

Kapabelt program

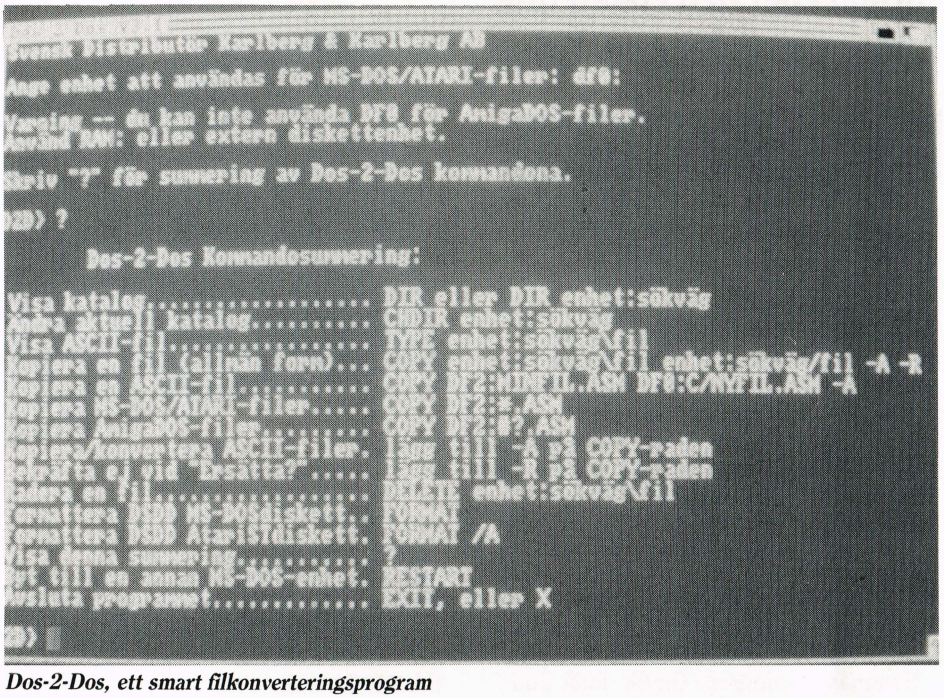
Vad exakt klarar då programmet av? Den version jag testade var den svenska versionen nummer 3.1., och den klarar av att hantera alla MS-DOS-enheter du har inkopplade om du har ett PC-kort i din 2000.

Det klarar av att hantera både enkel- och dubbelsidigt formatterade disketter i såväl 3,5 som 5,25-tumsformat. Dessutom kan man välja att omvandla Amigans tecken vid kopieringen för att passa IBM- standard.

En sak klarar dock programmet inte av, och det är att använda sig av den inbyggda 5,25-tums diskettenheten som hör till PC-kortet du stoppar i en Amiga 2000. Detta hör tydligen samman med konstruktionen på denna, externa enheter klarar programmet galant.

Kommandon

När du startar upp programmet täcks hela skärmen av ett fönster som påminner om vårt kära CLI/SHELL-fönster. Efter att ha specificerat vilken drive som skall användas som MS-DOS/Atari-enhet visas följande prompt:D2D



Dos-2-Dos, ett smart filkonverteringsprogram

DOS-2-DOS är då redo att ta emot något av de 9 kommandon som finns att tillgå.

Manualen

Den svenska översättningen som gjorts av Karlberg & Karlberg är föredömligt upplagd. På 32 sidor gås programmets funktioner genom på ett lätt och rakt sätt. Manualen, som framställts med hjälp av Professional Page, får alltså fullt godkänt, och jag tror att till och med min lillasyster skulle kunna använda programmet efter att ha läst denna!

Hur fungerar det då?

Ja, för alla som använt MS-DOS-datorer så är det inte speciellt svårt att få fart på programmet, men även för er som är någorlun-

Forts från sid 39

Snabbisar

Face Off

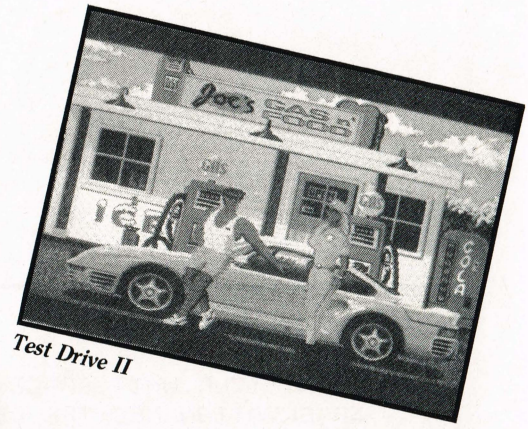
- som betyder tekning, är Ancos bud på hur ett ishockeyspel till Amigan ska se ut. Tyvärr tar de sig inte ens ur egen försvarszon. Det är nu en gång så, att när man väl sett ett riktigt bra spel av en typ (i det här fallet Wayne Gretzky), blir andra spel av typen automatiskt jämförda med det. Och den jämförelsen pallar Face Off inte för. Inte på långa håll...

Face Off ligger handlingsmässigt närmare C64-spel från 1983 än moderna Amigaspel. Det är inte ishockey, snarare en slags shoot'em up. Passningspel existerar knappast, det hela blir mest solokörning och skott på mål.

Planen scrollar uppåt och nedåt, med ditt mål i nedre delen. Grafiken vinner knappast några skönhetspris, det roligaste är den hejarklack som signalerar målen.

När Anco nu med Kick Off visat att de faktiskt kan göra riktigt bra spel, finns det inga ursäkter för ett spel som Face Off längre. Skärpning, Anco!

Nej, det här är rent ut sagt ett eländigt spel. Nog om detta!
Totalbedömning: 11%



Test Drive II

Test Drive II - The Duel

Du kan bara inte ha missat Test Drive till Amigan, det första bilspelet av klass till maskinen. Och nu kommer uppföljaren, som vid första anblicken ser precis likadan ut. Vid andra anblicken visar det sig dock att spelet förbättrats i alla avseenden.

Nu kan du välja svårighetsgrad, vägarna är nya och bättre med bl a tunnlar, kraschsektionen när du kör över ett stup eller ut i vattnet är bättre, grafiken är fräckare och mer detaljerad. Dessutom finns det tilläggsdisketter med fler bilar och fler vägar som man kan köpa till.

Det lustiga med spelet är att det egentligen är ganska enformigt, men att det ändå förmår att hålla intresset uppe. Varför? Tja, svårt att säga, men just förmågan att hålla spelaren fängslad är ju det som kännetecknar ett bra spel. Och The Duel ÄR ett bra spel.
Totalbedömning: 73%

---

da insatta i CLI:n så borde det inte vara några större problem.

Vi kan ta ett litet exempel för att visa hur lättanvänt programmet är. Vi tänker oss att vi ska föra över en ordbehandlingsfil, vilken som helst. (Fast i det här fallet faktiskt just den här texten du läser). Den ligger då alltså på en diskett med ett underbibliotek som egendomligt nog heter ARTIKLAR. Filen heter ju då naturligtvis DOS2DOS, du startar då upp programmet genom att dubbelklicka på ikonen. Fönstret öppnas upp och programmet frågar efter vilken enhet du vill använda som MS-DOS-enhet. Du svarar då snabbt och elegant DF0: och har därigenom plötsligt en helt ny MS-DOS-enhet monterad!

Formattering först

Eftersom du som vanligt inte har några MS-DOS-formatterade disketter hemma måste du nu formattera en diskett. Du stoppar då in den i DF0:, skriver FORMAT och trycker return.

När formatteringen är klar är det sedan dags för själva kopieringen. Stoppa in disketten med artikeltexten i DF1: och skriv sedan CHDIR DF1:. Därefter skriver du COPY ARTIKLAR/DOS2DOS DF0: -A och trycker return. Nu kommer det häftiga, din nyskrivna artikel läses av, översätts till IBM-teckenuppsättning och kopieras över till den nyformatterade MS-DOS-disketten.

Mycket praktisk om man har en PC på jobbet med modem till, bara att smyga dit och skicka iväg filen utan att behöva betala telefonräkningen själv!

Batchfiler

En snitsig finess är att du även kan ange kommandoparametrar direkt från CLI när du anropar DOS- 2-DOS. Då blir det ju enkelt att skriva små batch-filer för att utföra kopieringar som återkommer. (En batch-fil är alltså en sådan där du anropar med EXECUTE från CLI). Om du alltid lägger dina artiklar på samma ställe skriver du alltså en liten batch-fil med följande enkla innehåll : D2D DF0: COPY DF1:ARTIKLAR/#? DF0: -A

Kalla denna filen för till exempel TRANSFER. Med kommandot EXECUTE TRANSFER kommer sedan alla artiklar på disketten i DF1: att konverteras och läggas över på MS-DOS-disketten i DF0:.

Slutsats

För er som har användning av ett program för överföring mellan Amiga, Atari och/eller MS-DOS finns ingen anledning att tveka. DOS-2-DOS uppfyller alla krav som kan ställas på ett program av denna typ. En förutsättning är dock att du har två diskettenheter till ditt förfogande, eftersom programmet låser en enhet.

Du kan i och för sig först kopiera över filerna till RAM-disken innan du startar upp programmet, men då har du ödat lite tid till. En extra diskettenhet är dock så nyttig att alla seriösa användare borde ha en, så det bör ju inte vara något problem.

Kostnaden, 695 kr inklusive vår kära moms, är ju ett lågt pris att betala för att slippa hålla på att kladda med modem och annat elände för överföringen. Om inte annat så kostar ju telefonen också en slant, om ni inte tror mig kan ni ju fråga min gode kollega Pontus Lindberg vad 9990 markeringar på ett kvartal går på. ■

Checkout:
Karlberg & Karlberg, tel 046-474 50.
Pris 696:- inkl moms

SUPER 20

Vill du vinna 500 riksdaler skattefritt? (Vem vill inte det?). Sätt dig då vid datorn och tillverka ett program - ju bättre det är desto större chans har du på de 500 kronorna. Men 300 och 100 kronor är ju inte fy skam det heller! Endda villkoret - högst 20 rader!

```
10 PRINT"(CLR,WHIT,SPACE)LINJ
   ÄR EKVATION MED TRE OBEKANTA"
20 PRINT"(CRSR DWN,SPACE3)AV TYPEN:"
30 PRINT"(CRSR DWN,SPACE)A1X+B1Y+C1Z
   =D1"
40 PRINT" A2X+B2Y+C2Z=D2"
50 PRINT" A3X+B3Y+C3Z=D3"
60 INPUT"(CRSR DWN)A1";A1
   :INPUT"B1";B1:INPUT"C1";C1
   :INPUT"D1";D1
70 INPUT"A2";A2:INPUT"B2";B2
   :INPUT"C2";C2:INPUT"D2";D2
80 INPUT"A3";A3:INPUT"B3";B3
   :INPUT"C3";C3:INPUT"D3";D3
90 K=(A1*B2*C3+B1*C2*A3+C1*B3*A2-C1*
   B2*A3-B1*A2*C3-A1*B3*C2)
100 X=(D1*B2*C3+B1*C2*D3+C1*B3*D2-C1
   *B2*D3-B1*D2*C3-D1*B3*C2)/K
110 Y=(A1*D2*C3+D1*C2*A3+C1*D3*A2-C1
   *D2*A3-D1*A2*C3-A1*D3*C2)/K
120 Z=(A1*B2*D3+B1*D2*A3+D1*B3*A2-D1
   *B2*A3-B1*A2*D3-A1*B3*D2)/K
130 PRINT"X=";X:PRINT"Y=";Y
   :PRINT"Z=";Z
```

500.-

Programnamn: Ekvationslösare
Datortyp: C-64

Vid lösning av ekvationer med flera obekanta så krävs långa och tråkiga uträkningar. Därför är detta program till nytta vid sådana tillfällen. Exempel:
2x-y+4z=12
6x-2y-2z=8
x+y-4z=6

Resultat: X=6 Y=11,2 Z=2,8

Enkelt, eller hur?

Insänt av:
Ruben Ilahue
c/o Göthe
Munkarpsv. 13b
243 32 Höör

Programnamn: STOR
Maskintyp: Amiga

Det här är en förbättring av UCASE$-funktionen. Det är nämligen så att UCASE$ inte fungerar på ä,å och ö. Detta kan avhjälpas med detta program. Så mycket mer finns det inte att säga om denna lilla rutin.

Insänt av:
Per-Olof Persson
Kyrkåsvägen 6
416 56 Göteborg

Programnamn: Studs
Maskintyp: Amiga

De flesta av de program till Amigan som kommer in till Super20 har något med grafik att göra, så även detta program. Nå, vad gör då det här programmet för grafiska konster? Jo, i sitt grundutförande ritar det upp 30 linjer som studsar runt. Detta kanske inte låter så upp hetsande men det ger upphov till en hel del vackra mönster och effekter.

Programmet kan lätt ändras så att det blir ännu lite roligare, om man vill kan man t.ex. ändra på varia-

100.-

```
r(1)=.41:r(2)=.99:g(1)=.12:g(2)=.34
b(1)=.67:b(2)=.99:p=3:COLOR 3,2
PRINT  TAB (17);"This is color flip"
cykla
END
SUB cykla STATIC:SHARED r(),g(),b(),p
s=.01:i=1:d=2
flip:
FOR a=r(i) TO r(d) STEP s:PALETTE p,a,g(i),b(i):GOSUB key:NEXT
FOR c=g(i) TO g(d) STEP s:PALETTE p,a,c,b(i):GOSUB key:NEXT
FOR n=b(i) TO b(d) STEP s:PALETTE p,a,c,n:GOSUB key:NEXT
IF s=.01 THEN s=-.01:SWAP i,d:GOTO flip
IF s=-.01 THEN s=.01:SWAP i,d:GOTO flip
key:
IF INKEY$=""THEN RETURN
PALETTE p,r(1),g(1),b(1)
END SUB
```

300.-

Programnamn: Color flip
Maskintyp: Amiga

Här har vi en fin liten subrutin som heter colorflip. Denna rutin gör en cykel mellan två färger som du själv bestämmer. De första raderna i programmet är ett demo på hur själva rutinen funkar. Dessa behöver man allså inte ha med.

Om du nu ska använda rutinen i ett eget program så måste du först skriva in en del variabler, det ser ut som följer:

r(1)=röd1 g(1)=grön1
b(1)=blå1

r(2)=röd2 g(2)=grön2
b(2)=blå1

p= vilken palett som ska cyklas.

Viktigt är att värdena r(1),g(1) och b(1) är mindre än vad r(2),g(2) samt b(2) är. Ett tips när du använder programmet är att du skriver in variablerna i början. Sedan är det bara att skriva cykla när man tycker att det behövs. Datorn slutar att cykla när en tangent trycks ner och återgår till huvudprogrammet.

Insänt av:
Henrik Jönsson
Nikolaigatan 12
21421 Malmö

```
start:PRINT  :INPUT "Skriv ett ord",x$
  u$= UCASE$(x$)
  STOR(x$)
 PRINT "UCASE$(x$)";u$,"STOR$";x$:GOTO start
SUB STOR(x$) STATIC
  SHARED x$
  f$="":FOR k=1 TO LEN (x$)
  d =ASC(MID$(x$,k,1)): IF d >96 AND d<123 THEN d=d-32:GOTO sam
  IF d=229 THEN d=197:GOTO sam
  IF d=228 THEN d=196:GOTO sam
  IF d=246 THEN d=214:GOTO sam
 sam:f$=f$+CHR$(d):NEXT k:x$=f$ END SUB
```

100.-

beln n som står för antalet linjer som skall ritas upp innan den sista suddas ut eller xm, ym som är hastigheten på linjens ändar.

Något som jag tyckte var kul var att stegvis öka eller minska antalet linjer som ritas upp. Det gör man enklast genom att lägga till en rad strax innan next-satsen. Raden kan se ut som den här: n=n+.12, eller något liknande. Det viktigaste är att det tal man adderar till n ligger mellan 0 och 1. Den sista effekten jag tänkte tipsa om är att man kan sätta dit ,b eller ,bf i slutet av de båda LINE raderna. Då får man nämligen rektanglar i stället för linjer.

Insänt av:
Erik Spåre
Ekhammar ärna
633 47 Eskilstuna

```
SCREEN 1,325,240,5,1:WINDOW 1,"STUDS",(0,0)-(302,217),15,1
CLEAR,90000&
DIM xd1(5000),xd2(5000),yd1(5000),yd2(5000)
f=11:x1=20:x2=200:y1=55:y2=129:xm1a=4:xm1b=-4:xm2a=-4
xm2b=4:ym1a=4:ym1b=-4:ym2a=-4:ym2b=4:i1=1:i2=-1:n=30
PALETTE 0,0,0,0:PALETTE 1,0,0,0:PALETTE 2,0,0,0
FOR e=11 TO 30:a=(e-10)/20:PALETTE e,0,0,a:NEXT
FOR v=1 TO 5000
  x1=x1+xm1a:x2=x2+xm2a:y1=y1+ym1a:y2=y2+ym2a
  IF x1<2 OR x1>300 THEN SWAP xm1a,xm1b
  IF y1<2 OR y1>215 THEN SWAP ym1a,ym1b
  IF x2<2 OR x2>300 THEN SWAP xm2a,xm2b
  IF y2<2 OR y2>215 THEN SWAP ym2a,ym2b
  xd1(v)=x1:xd2(v)=x2:yd1(v)=y1:yd2(v)=y2
  f=f+i1:IF f>29 OR f<12 THEN SWAP i1,i2
  LINE (x1,y1)-(x2,y2),f
  IF v>n THEN LINE (xd1(v-n),yd1(v-n))-(xd2(v-n),yd2(v-n)),0
NEXT v:STOP
```

Programnamn: Expander
Datortyp: C-64

Detta program förstorar upp valfri teckenuppsättning två gånger i varje/valfri led. Ladda in teckenuppsättningen på adress 12288 ($3000 hex). SYS 49152 expanderar sedan i X-led, och SYS 49281 expanderar i Y-led.

Programmet expanderar endast de 64 första tecknena, eftersom det inte får plats fler i teckenminnet, om man expanderar i båda leden. För att skriva ett tecken som är förstorat i X-led gör man på följande vis:

Tryck in tangenten för tecknet i fråga, och den vänstra halvan av tecknet visar sig på skärmen. Ta skärmkoden för tecknet (peek(1024+rad*40+kolumn)), addera med 64, och där är skärmkoden för den högra halvan. Man kan också använda SHIFT-tangenten för att addera 64, så att man får fram tecknets högra halva.

För att skriva ett tecken expanderat i båda leden, dvs fyra rutor på skärmen, skriv undre delen på samma sätt som ovan, fast med RVS ON. Vill du bara använda Y-led, är det praktiskt att flytta positionerna 13312-13823 till 12800, då kan du skriva tecknena precis som i X-led. Teckenuppsättningen aktiveras med POKE 53272,29 och stängs av med POKE 53272,21.

Insänt av:
Pierre Bulic'
Nilssonsberg 31
411 43 Göteborg

```
10 FOR I=49152 TO 49393:READ A:B=B+A
   :POKE I,A:NEXT
20 IF B<>32132 THEN PRINT"FEL I DATA"
   :STOP
30 DATA 169,48,141,12,192,169,0,141,
   11,192,173,0,48,133,2,41,15
40 DATA 133,251,165,2,41,240,133,254,
   162,0,24,102,251,32,107,192,102
50 DATA 252,32,119,192,102,252,24,38,
   254,32,107,192,38,253,32,119,192
60 DATA 38,253,232,224,4,208,225,173,
   11,192,141,81,192,141,86,192,174
70 DATA 12,192,142,82,192,232,232,
   142,87,192,165,253,141,0,48,165,
   252
80 DATA 141,0,50,238,11,192,208,3,
   238,12,192,173,12,192,201,50,240
90 DATA 3,76,10,192,96,144,5,169,1,
   133,2,96,169,0,133,2,96
100 DATA 165,2,201,1,240,2,24,96,56,
   96,169,0,141,142,192,169,48
110 DATA 141,143,192,162,0,189,0,48,
   157,241,192,232,224,8,208,245,174
120 DATA 142,192,142,195,192,142,204,
   192,232,142,198,192,142,207,192,
   174,143
130 DATA 192,142,196,192,142,199,192,
   232,232,232,232,142,205,192,142,
   208,192
140 DATA 162,0,160,0,185,241,192,157,
   0,48,157,1,48,185,245,192,157
150 DATA 0,52,157,1,52,200,232,232,
   192,4,208,231,173,142,192,24,105
160 DATA 8,141,142,192,144,3,238,143,
   192,173,143,192,201,52,240,3,76
170 DATA 139,192,96,24
```

SUPER 20

```
1 BA=49152:B=BAS/256
2 IF B<>INT(B)THEN PRINT"FELAKTIG BA
  SARESS":END
3 FOR I=0 TO 159
4 READ A:CS=CS+A
5 IF A=-1 THEN A=B
6 POKE BA+I,A
7 NEXT
8 IF CS<>19605 THEN PRINT"GURU MEDIT
  ATION... FEL I DATA!":END
9 DATA 76,18,-1,76,12,-1,76,25,-1,76,
  34,-1,169,198,162,255
10 DATA 208,23,169,230,162,1,32,41,
  -1,173,17,208,41,239,141,17
11 DATA 208,96,173,17,208,9,16,208,
  245,134,254,141,146,-1,162,0
12 DATA 142,21,208,120,169,127,141,
  13,220,169,241,141,26,208,173,25
13 DATA 208,141,25,208,169,43,141,18
  208,173,17,208,41,96,133,255
14 DATA 169,111,162,-1,32,103,-1,166
  254,208,252,120,142,26,208,202
15 DATA 142,13,220,169,49,162,234,
  141,20,3,142,21,3,88,96,169
16 DATA 1,141,25,208,166,254,160,7,
  136,208,253,234,234,234,172,18
17 DATA 208,136,152,41,7,9,16,141,17,
  208,202,208,233,5,255,141
18 DATA 17,208,198,254,208,7,169,27,
  5,255,141,17,208,76,188,254
```

Programnamn:
Screen-scroll
Datortyp: C-64

Denna rutin är enkel. Den scrollar helt enkelt skärmen antingen upp eller ner. Man kan också släcka och tända skärmen, vilket kan vara användbart. Kommandona är:
SYS 49152 Scrolla ner+släck skärmen.
SYS 49155 Tänd skärmen+scrolla upp.
SYS 49158 Släck skärmen.
SYS 49161 Tänd Skärmen.

Programmet fungerar också på högupplösningsbilder, och kan användas i spel och slideshows.

Insänt av:
Andreas Löfwenmark
Hökgatan 2
566 00 Habo

Så här deltager du!

För att vinna våra skattefria premier, ska du uppfinna världens bästa rutin på 20 linjer eller mindre, och skicka in den till oss. Vi vill varken ha band eller diskett med rutinen på, bara en läslig utskrift, antingen den är handskriven eller från skrivare.
Våra experter testar alla inkomna rutiner, och den lyckliga vinner 500 kr. Till andrapristagaren blir det 300 kr, och en hundralapp väntar övriga publicerade bidrag. Vi skickar inte utskriften tillbaka, och kan vi inte använda den "arkiverar" vi den bara lodrätt.
Du har ingen möjlighet att ringa och kontrollera om just din rutin blivit utvald, eftersom urvalet sker i sista ögonblicket före deadline. På så sätt kommer de allra bästa rutinerna våra läsare tillgodo.
Alltså, skicka in och vinn till adressen:

Oberoende COMputer
St. Kongensgade 72
DK-1264 Köpenhamn K
Danmark
Märk kuvertet "SUPER 20/(vilken typ av maskin du har)"
OBS! Det är olagligt att stjäla program från andra tidningar och skicka in dem som om de vore ens egna. Överträdelse beivras!

Maskinkod 64

Dags igen för en ny artikel i vår evighetslånga artikelserie om brödburkens programmering. Vår skribent har denna gång slitit sitt hår för att tillfredställa er, kära läsare!

Av Jonny Bergdahl

Visst är det väl så att telefonen ringer varje gång man precis krupit ner i badet. Avbruten av den gälla signalen sveper man sitt badlakan om sig och klafsar fram till telefonen bara för att upptäcka att det är någon som ringt fel. Och precis när man återvänt till sitt varma sköna bad, så ringer det igen. Upp och svara, suckande och svärande.

Vad har det här med datorer att göra säger du då? Ja, det här var väl en bra illustration av hur det här med interrupt fungerar!? Nu är det nämligen dags att komma in på mer avancerade övningar i vår maskinkodsserie. I det här avsnittet ska vi gå igenom hur man sätter upp ett rasterinterrupt.

Videochipet

För att starta ett rasterinterrupt används tre olika register i videochipet. Rasterregistret ligger på adress $D012, och har två olika ändamål. När man läser registret får man nuvarande rasterrad. När man skriver till registret, däremot, så sparas detta värde för att jämföras med rastervärdet. När dessa är lika sätts bit 0 i interruptregistret på adress $D019. Nu kommer själva finessen.

Om samma bit är satt i interruptmaskregistret på adress $D01A, så kommer även ett interrupt att genereras. Ytterligare en adress används, och det är adress $D011, som innehåller den nionde biten i rasterregistret. Det är bit 7 som används till detta.

Andra interrupt

Adress $D019, interruptregistret, har följande funktioner: Bit 0 jämför rasterraden, bit 1 kollar kollisioner mellan sprite och tecken, bit 2 kollar kollision mellan sprites och bit 3 kollar ljuspennan. Bit 7 talar om att ett interrupt har genererats.

För att ett interrupt ska ges så fordras att motsvarande bitar är satta i interruptmaskregistret på adress $D01A. För att rensa ett interrupt skriver man en etta till den bit man önskar på adress $D019, och eftersom övriga bitar inte påverkas så kan man på det här sättet ta hand om interrupt av fler källor på en gång.

Tilläggas bör att bitarna i $D019 sätts oavsett om maskregistret tillåter interrupt, detta för att man skall kunna läsa av dessa även utan att använda sig av interrupt.

Hoppvektorerna

För att kunna använda en maskinkodsrutin ihop med BASIC fordras att man har både Kernal och BASIC-romen inkopplade. Detta innebär att man blir tvungen att använda dom hoppvektorer som 64:ans Kernal tillhandahåller. När det gäller interrupt så ligger dessa på adress $0314 och $0315. Som vanligt ligger adresserna i ordningen lågbyte-högbyte. För att din 64:a ska hitta fram till ditt eget interrupt fordras alltså att du pekar ut din rutin med hjälp av dessa adresser.

CIA-chipet

När man går in på interrupt så brukar man alltid få slåss med CIA-chipet. Detta chip sköter om dom normala interrupten i din 64:a. 60 gånger i sekunden blir det ett interrupt som tar hand om att klockan går rätt, läser av tangentbordet och lite annat i den stilen.

Om man låter detta fortsätta inträffar det intressanta att två olika chip genererar interrupt samtidigt. Det innebär ju i sig inte något allvarligt, eftersom 64:an hanterar alla interrupt i den ordning dom kommer. Finessen med rasterinterrupt är ju att man kan synkronisera sina program med skärmen och av denna anledning behöver man alltså stänga av CIA-chipet. Det är räknare A i CIA-chip nummer 1 som står för de vanliga interrupten, och dessa stänger man enklast av genom att lägga värdet 0 i adress $DC0E, kontrollregister A för detta chip.

Registrens innehåll?

I vanliga fall måste man spara undan innehållet i alla register innan man hoppar till sin egen interruptrutin, men eftersom vi här använder oss av dom hoppvektorer som finns, så har Kernal redan skött om den saken åt oss. För att slippa att lägga tillbaks värden i registren efter vår rutin så låter vi Kernal ordna även det. Här har vi två alternativ att välja på.

Antingen hoppar vi till den normala interruptrutinen, vilket får datorn att fungera normalt ihop med vår rutin, eller så låter vi Kernal enbart ta hand om städningen av registren. Normalt innehåller interrupthoppvektorerna adressen $EA31, och det är denna adress vi skall hoppa till om vi vill ha tillgång till tangentbord osv. Visserligen kommer klockan att gå fel, men hur många använder sig av den ändå, den går ju fel när du laddar från disk också.

För att bara lägga tillbaka registervärden hoppar vi istället till slutet på den normala interruptrutinen, till adress $EA81.

Programexemplet

För att illustrera användningen av rasterinterrupt har jag kokat ihop ett enkelt exempel. För att inte krångla till det för mycket i början, så har jag gjort ett program som delar upp skärmen i två halvor. Den övre halvan visar skärmens tecken som stora bokstäver och grafik, medan den nedre halvan visar både små och stora bokstäver. För ni lättare ska kunna se skarven har jag gjort så att skärmen har en annan färg i de olika partierna.

Eftersom det är ett väldigt kort program så kan jag ta och förklara funktionen ganska grundligt. Del 1 som ligger på adress $C000 startar upp själva rasterinterruptet och pekar ut interruptrutinen som ligger på adress $C100. Vi börjar som vanligt med att stänga av det normala interruptet en kort stund medan vi aktiverar vårt nya. Sedan följer i tur och ordning uppläggning av hoppvektorerna och inläggning av rasterrad i rasterregistren. Maskregistret sätts till att tillåta interrupt från rasterregistret, och CIA-chipet stängs av. Slutligen lägger vi in ett startvärde för vår interruptrutin.

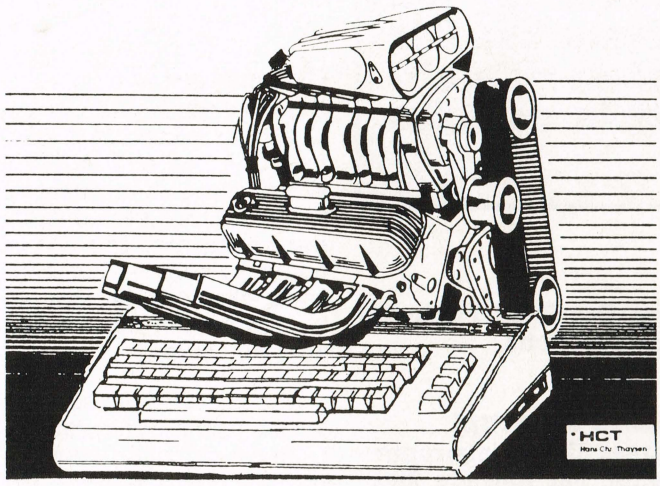
Om du vill ha en stunds fördröjning innan rutinen börjar jobba kan du byta ut $02 mot till exempel $80 eller $FF. Detta beror på att adress $02 räknas ner och kollas mot 0 för att avgöra om det är övre eller undre delen som är aktiv.

Del 2

Själva interruptrutinen, som börjar på adress $C100, börjar med att rensa interruptregistret. Detta gör man enklast genom att skriva samma värde man läst från detta register.

Om man kör flera olika interrupt, däremot får man vackert ta och maska ut rätt bit att skriva till för att inte hoppa över något. Sedan sätter vi ett nytt värde för rasterjämförelse, och här får vi inte glömma bort rastrets nionde bit. Sedan gör vi taktisk fint. Vi lägger in $15 i A-registret innan vi testar om det är övre eller undre delen som gäller.

På raden under minskar vi vår räknare och om det var okej hoppar vi ner till raden där A-registret sparas. Skulle jämförelsen vara lika så får däremot A-registret ett annat värde innan denna rad, tillsammans med rasterregistret, som sätts till att peka på sista skärmraden. Sedan lagras alltså A-registrets värde i adress $D018, och det är denna adress som styr vilken teckenuppsättning som skall vara aktiverad. $15 ger stora bokstäver plus grafik, medan $17 ger både små och stora bokstäver. För att vi ska se skarven ordentligt så lägger vi detta värde även i ramens färgregister.



Till sist tar vi till en taktisk fint till. Vi stänger av CIA-chipet igen, denna gång i interruptet. Detta är bara nödvändigt när man använder en cartridge av den nyare typen, till exempel Action Replay Mk IV, som envisas med att slå på CIA-chipet hela tiden när den ligger i editorläge. I programläge, däremot, till exempel om du kör ett BASIC-program samtidigt med ditt interrupt, så behövs inte detta. Slutligen avslutar vi rutinen med ett hopp till adress $EA31 som sköter om tangentbord och sådant åt oss, tillsammans med återställning av registren.

Nästa gång

- ska vi titta lite mer på hur man använder sig av olika interrupt, och lite längre fram är det dags att avsluta den här serien med att visa hur man gör ett enkelt spel. Men tills dess kan ni ju ta och försöka köra några av mina gamla programexempel på interrupt, till exempel färgbalkarna i nummer 5-89. Jag vill också passa på att tacka Anders Hermansson för att han lånat ut sin dator till mig när jag kraschat min med hjälp av en hemkörd parallellkabel. Nästa nummer blir det ingen maskinkodskurs, eftersom militärmyndigheten i sin oändliga visdom beordrat undertecknad att inträda i statens tjänst några veckor.

Ut i skogen och panga, det är ju så man blir avundsjuk på er som får sitta inne i värmen och programmera...

Programexempel

Del 1 - Startrutin

```
C000    SEI           ;STÄNG AV INTERRUPTEN
C001    LDA #$00      ;PEKA UT INTERRUPTRUTINEN
C003    STA $0314     ;VIA INTERRUPTVEKTORERNA
C006    LDA #$C1      ;PÅ ADRESS $0314 & $01315
C008    STA $0315
C00B    LDA $D011     ;HÄMTA IN REGISTER FÖR
C00E    AND #$7F      ;RASTRETS HÖGBIT OCH SÄTT
C010    STA $D011     ;DEN TILL 0.
C013    LDA #$81      ;SÄTT INTERRUPT PÅ RAD $81
C015    STA $D012
C018    LDA #$01      ;GÖR KLART FÖR INTERRUPT
C01A    STA $D01A     ;I MASKREGISTRET
C01D    LDA #$00      ;STÄNG AV NORMALA (CIA#1)
C01F    STA $DC0E     ;INTERRUPTET.
C022    LDA #$01      ;STARTVÄRDE FÖR INTERRUPT-
C024    STA $02       ;RUTINEN.
C026    CLI           ;STARTA UPP INTERRUPT-
                      ;RUTINEN
C027    RTS           ;OCH ÅTERVÄND.
```

Del 2 - Interruptrutinen

```
C100    LDA $D019     ;ÅTERSTÄLL INTERRUPTET GENOM
C103    STA $D019     ;ATT SKRIVA SAMMA VÄRDE.
C106    LDA #$81      ;RASTERRAD FÖR NÄSTA INTERRUPT
C108    STA $D012
C10B    LDA $D011     ;ÅTERSTÄLL ÄVEN HÖGSTA BITEN
C10E    AND #7F       ;FÖR RASTERREGISTRET
C110    STA $D011
C113    LDA #$15      ;VÄRDE FÖR STORA BOKSTÄVER
C115    DEC $02       ;MINSKA RÄKNAREN
C117    BNE $C124     ;HOPP OM DET SKA VARA STORA
                      ;BOKSTÄVER.
C119    LDA #$02      ;ÅTERSTÄLL RÄKNAREN
C11B    STA $02
C11D    LDA #$FA      ;NY RAD FÖR NÄSTA RASTER-
C11F    STA $D012     ;INTERRUPT
C122    LDA #$17      ;VÄRDE FÖR SMÅ BOKSTÄVER.
C124    STA $D018
C127    STA $D020     ;SAMMA VÄRDE FÖR RAMFÄRG.
C12A    LDA #$00      ;STÄNG AV CIA-CHIPET IGEN
C12C    STA $DC0E     ;(FÖR CARTRIDGE-ÄGARE)
C12F    JMP $EA31     ;HOPPA TILL NORMALA
                      ;INTERRUPTET.
```

Panasonics skrivare har funnits på marknaden ett tag, men de har alltid verkat tämligen anonyma och har hamnat i skuggan av STAR:s utmärkta skrivare LC 10 och LC 24-10. Finns det någon anledning till detta eller är marknaden bara orättvis?? Oberoendes utsände presenterar här sina råd och rön...

Billig och bra trycksak

Panasinic KX-P1081 som är en 9 nålars skrivare i 2000 kronors klassen.
Tryck kavalitén framgår av dessa exempel.....

Programstyrd NLQ.....

Panasinic KX-P1081 som är en 9 nålars skrivare i 2000 kronors klassen.
Tryck kavalitén framgår av dessa exempel.....

Standard.....

Panasinic KX-P1081 som är en 9 nålars skrivare i 2000 kronors klassen.
Tryck kavalitén framgår av dessa exempel.....

NLQ.....

Panasinic KX-P1081 som är en 9 nålars skrivare i 2000 kronors klassen.
Tryck kavalitén framgår av dessa exempel.....

Compressed.....

Panasinic KX-P1081 som är en 9 nålars skrivare i 2000 kronors klassen.
Tryck kavalitén framgår av dessa exempel.....

Grafik.....

Skrivare finns numera i väldigt många prisklasser. De billigaste ligger på cirka 2000 kronor och här återfinns bland andra Commodores egen MPS 1230 och även Panasonics KX-P1081. Bland de lite dyrare, drygt 3000 kronor, finns t.ex. Star LC 10.

Det är just de nämda jag tänker försöka relatera till då jag tittar närmre på KX:en.

Början

Att testa en printer är lättast att börja göra med en självtest. Som på de flesta andra printrar håller man nere LF då man slår på strömmen. KX:en börjar då spruta ut en massa rader text med lite förskjutning. Redan vid denna första anblick fick

Av Pontus Lindberg

Billig och bra

jag en uppfattning om i vilken klass utskriftskvaliteten ligger. Den är klart bättre än MPS 1230 men klassen sämre är LC-10. Detta är ett intryck som består under hela testet och som jag fortfarande har kvar.

Finesser

KX:en klarar av både friktions- och traktormatning. Detta är något man väl förväntar sig av alla printrar, men är ändå värt att påpeka. En liten invändning på ett tidigt stadium är att Panasonics traktormatning känns lite "sladdrig" på grund av att traktorpiggarna sitter på en aningen vinglig plastremsa som är mer av typen larvfötter på en pansarvagn än de runda traktorhjul som t.ex. Star LC-10 har.

Jag saknade vidare den anordning som vanligen styr ner enskilda ark i skrivaren så att de alltid kommer rakt. Detta har till och med MPS 1230 som standard.

Reglage

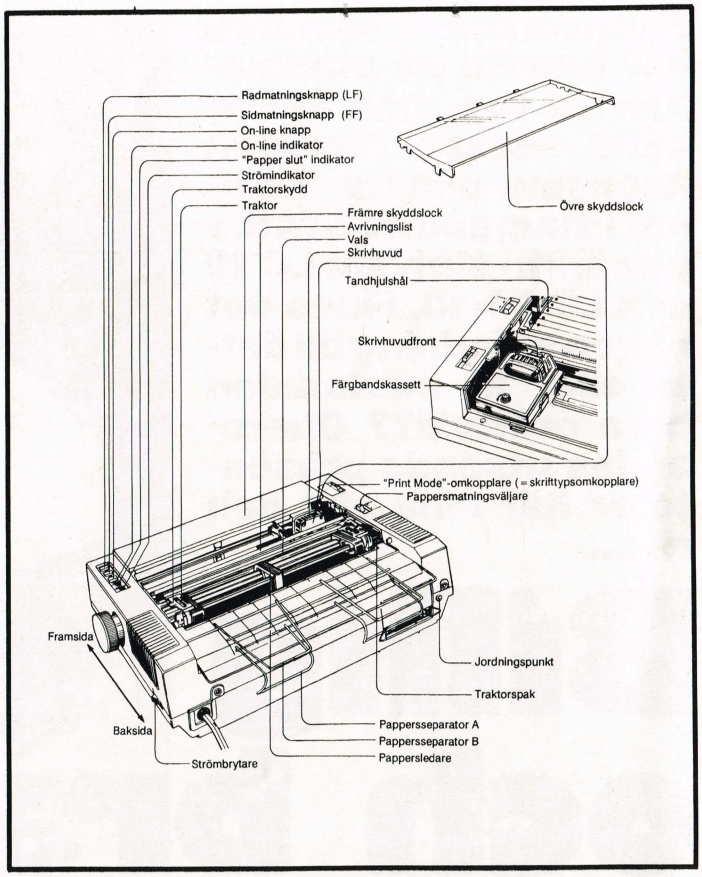
Alla KX:ens relage är rejäla och enkla att hantera. Växling mellan traktor/friktion, växling mellan standard/NLQ/compressed skrivläge samt de tre knapparna man altid hittar, nämligen "Line Feed", "Form Feed" och "On Line". DIP-switcharna sitter rakt ned i lådan och kan nås direkt utan att man måste jonglera med skrivaren för att se bak- eller undersida som man tidigare alltid fick göra på äldre skrivare. En liten genomskinlig plastskiva skyddar dem utan att förhindra insyn. Genomgående bra och rejält med andra ord.

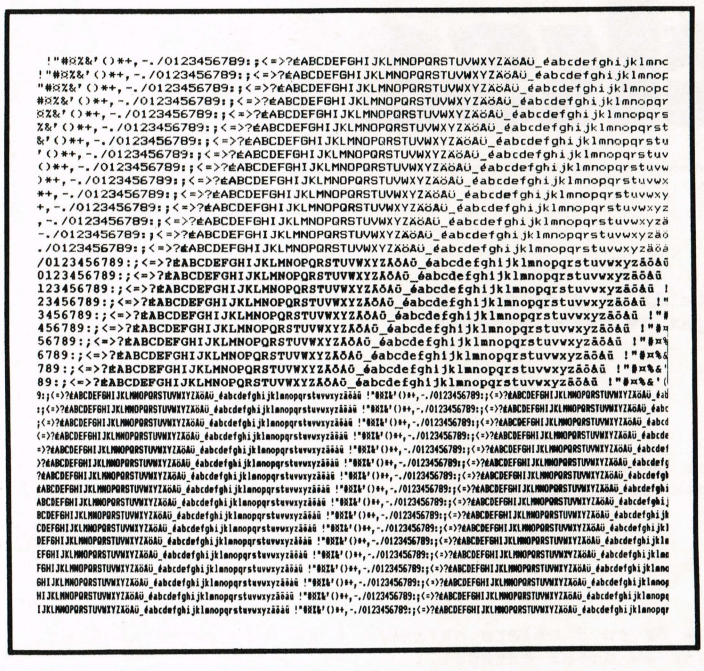
Hastighet

De flesta printrar skriver ut ca 120 tecken per sekund eller strax däröver och KX:en är inget undantag. 120 TPS i "draft pica" och 144 i "draft elite". Den inbyggda bufferten på 1024 tecken, d.v.s. 1 K, gör att utskriften blir aningen effektivare då man inte måste vänta på att skrivaren skall bli riktigt klar. Givetvis finns det logisk sökning som innebär att den skriver åt båda hållen, inte skriver på rader som bara består av spaces och så vidare.

Färgbandet

Färgbandet påminner starkt om alla andra färgband, främst på grund av att man kommit på en ny standard av färgkassett. Det verkar inte finnas två skrivare som använder sig av samma färgband förutom





Citizen 120D och MPS 1250. Detta har dock en liten inbyggd bläckpatron som kan färga bandet och på så vis förlänga bandets livslängd med säkert en miljon tecken.

Manualen1

Två manualer får man med i paketet varav en engelsk och en svensk. Jag läste givetvis den något tunnare svenska och kunde snabbt konstatera att det var en föredömligt utförd manual med korta, informativa textstycken mellan talrika och överskådliga illustrationer. Klart positivt med andra ord.

Anslutningar

Kabeln man ansluter är en helt standardiserad Centronics parallellkabel och andra änden går direkt in i min Amiga. Har man en 64 får man komplettera med ett interface, men det är ju 64/128-användarna vana vid att tvingas göra. Jag hade inget interface av denna typ så jag har bara provat att köra skrivaren tillsammans med Amigan.

Printerdriven som användes tillsammans med med ProWrite var EpsonX, som i WB 1.3 dessutom har suffixet "Commodore MPS 1250". Jag stötte inte på några som helst problem under de dagar jag hårdtestade Panasonicskrivaren oavsett om det var text eller bilder jag skrev ut, och det är ju en av huvudkriterierna när man letar efter en bra skrivare.

Sammanfattning

Kvalitetsmässigt hamnar utskrifterna på KX-P1081 mellan MPS 1230 och LC-10. Att köpa en MPS 1230 då man har en prismässigt jämnbördig KX att välja på verkar vara ett utpräglat hån mot de fria marknadskrafterna. Att jämföra KX:en mot den förvisso dyrare LC-10 blir däremot mer spekulativt. LC-10 är förvisso klart dyrare, men den presterar å andra sidan mycket över det man kan pressa KX:en till; LC:n har ju flera inbyggda fonter, två compressed modes, jämnare och mjukare utsktifter samt en betydligt enklare pappershantering. Kan man vara utan dessa kvaliteter är KX ett klart vettigt alternativ, även om jag nog ändå tycker att man får betydligt mer för pengarna då man väljer en STAR LC- 10.

Läserundersökning - skicka in och vinn!

■ Oberoende COMputer är en tidning i ständig utveckling, och eftersom vi hela tiden försöker göra tidningen så bra som möjligt vill vi gärna höra vad DU tycker om det hela. Det är ju trots allt du som ska läsa tidningen. Nu får du därför en enastående möjlighet att påverka kommande nummer av Oberoende COMputer. Fyll i nedanstående läsarundersökning med kryss och kommentarer och skicka in den till oss i ett slutet kuvert. Du får gärna skicka in en fotokopia av läserundersökningen om du inte vill klippa i tidningen! Bland de insända undersökningarna lottar vi ut hela 25 "tack-för-hjälpen"-priser. Och det är inte småpotatis du har chans att få. Vad sägs om följande prissamling: 5 st halvårsprenumerationer på Oberoende COMputer, 10 spel (Amiga eller C64), 10×10 BASF-disketter (3.5" eller 5¼"). Jo, det kan löna sig att skicka in, eller hur?

Skicka in läsarundersökningen i slutet kuvert till följande adress:

Oberoende COMputer
St. Kongensgade 72
DK-1264 Köpenhamn K
Danmark

Märk kuvertet "Läserundersökning".

OBS! Ditt bidrag måste vara oss tillhanda senast den 31/10 1989.

Tack på förhand för din medverkan!

Ålder: _____

Kön,
Man ☐ **Kvinna** ☐

Yrke:

_____

Skolutbildning:

_____

Upplysningar om din datorutrustning:

Har du en dator? JA ☐ NEJ ☐
Om JA, vilken typ?
Commodore 64 ☐
C16/Plus4 ☐
Amiga 500 ☐
Amiga 1000 ☐
Amiga 2000 ☐
Annan: _____

Om du har en Amiga, hur mycket RAM har den? _____ **Kb**
Har du skrivare?
JA ☐ NEJ ☐
Om JA, vilken typ?
Dot-matrix ☐
Typhjul ☐
Laser ☐

Har du datasette?
JA ☐ NEJ ☐

Hur många diskettestationer har du? _____

Har du modem?
JA ☐ NEJ ☐
Vad har du för skärm?
Färgmonitor ☐
Monokrom monitor ☐
TV ☐
Annan extrautrustning: _____

Hur mycket har din dataanläggning kostat?
Under 1000 ☐
1000-2000 ☐
2001-5000 ☐
5001-10000 ☐
10001-20000 ☐
över 20000 ☐

Hur mycket kommer du att investera i din anläggning under det närmsta året?
1000-2000 ☐
2001-5000 ☐
5001-10000 ☐
10001-20000 ☐
över 20000 ☐

Var använder du din dator?
Hemma ☐
Arbete ☐
Skola ☐

Hur lång tid per dag använder du din dator? _____ **timmar**

Använder du datorn oftast:
Ensam ☐
Tillsammans med andra ☐
Vilka program använder du?
Ordbehandling ☐
Spreadsheet ☐
Databaser ☐
Grafik ☐
Video ☐
Ljud ☐
Desktop Publishing ☐
Kommunikation ☐
Äventyrsspel ☐
Actionspel ☐
Simulatorspel ☐
Andra ☐

Var köper du vanligtvis datautrustning och program?
Postorder ☐
Butik ☐

Vilken affär/postorderfirma handlar du oftast hos?
Namn på firman: _____

_____

Vilka tre program använder du oftast?
1 _____
2 _____
3 _____

Om du programmerar själv, vilka språk använder du?
BASIC ☐
Pascal ☐
C ☐
Assembler ☐
Andra: _____

_____

Vad tycker du om följande artiklar i detta nummer?

Artikel	Mycket bra	Bra	Medel	Under medel	Dålig	Läste den inte
Puls						
Super 20						
Lär Dig att lösa...						
Com/Post						
Soft Check						
1541 tips						
Printertest						
CLI-Skolan						
It's filmtajm, folks!						
AmigaBASIC						
Dos-2-Dos						
Snabbisar						
Packer						
Få stora sprites...						

Vad vill du läsa om?
Amiga ☐
Commodore 64 ☐

Hur lång tid har du läst Oberoende COMputer?
_____ år

Hur tycker du tidningen utvecklat sig under det senaste året? Har den blivit:
Bättre ☐
Sämre ☐
Oförändrat ☐
Vet inte ☐

Hur många nummer av Oberoende COMputer läser du per år?
Under 5 ☐
5-10 ☐
11 ☐

Var får du tag på Oberoende COMputer?
Prenumererar ☐
Tidningshandlaren ☐
Lån ☐
Biblioteket ☐

Hur många läser ditt exemplar av Oberoende COMputer?
Antal personer: _____

Vilka andra datortidningar läser du?
_____
_____
_____
_____
_____

Vilka andra datortidningar köper du?
_____
_____
_____
_____
_____

Har du förslag till ändringar eller förbättringar av Oberoende COMputer? (Skriv)
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

Namn: _____

Adress: _____

Postnr: _____

Ort: _____

ARCADE ACTION AID

Efter en liten paus är det dags för lite fusk igen. Man kan ju inte spela efter reglerna alltid, eller hur?

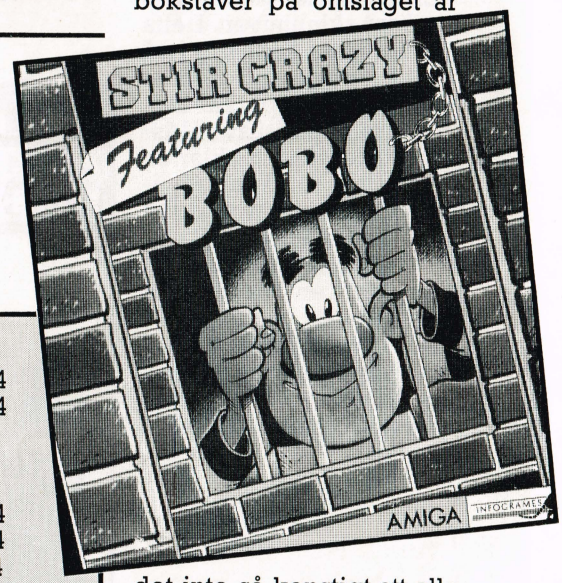
Samurai Warrior
Här ska du helt enkelt bara trycka ner F om du vill ha oändligt liv. Svårare än så var det inte!

West Bank
Joakim i Sjöbo har visserligen skaffat en Amiga 500, men han har för den skull inte glömt sin gamla brödburk. I West Bank ska man resetta och skriva
POKE 12713,165
för att få oändligt liv. Man startar igen med
SYS 4100
Allt enligt Joakim.

Army Moves
Samme man har också spelat Army Moves. Här ska man få nivå 5 genom att skriva 15863 när koden begärs.

The Last Ninja II
- tröttnar folk tydligen aldrig på. Det måste väl vara ett av de populäraste spelen genom tiderna, åtminstone om man ska döma av alla frågor och tips om spelet som kommit in till redax. Lars Olsson tipsar i alla fall om att man på sista delen sla kasta en stjärna på den vite samurajen så att han svimmar i mitten av den röda stjärnan. Då ska man tända alla ljusen varpå samurajen dör. Sedan går man och lägger tillbaka kristallkulan.

Bobo
Erik i Älvdalen har spelat det här grafiskt rätt fräcka spelet från franska Infogrames. Ja, egentligen heter spelet faktiskt Stir Crazy featuring Bobo, men eftersom Bobo står med störst bokstäver på omslaget är



det inte så konstigt att alla bara kallar det för Bobo.

Nåväl, ett tips från Erik! I avdelning 4, den med studsmattan, ska man ställa studsmattan under ett fönster så att fångarna studsar rätt upp. Stå så tills fyra fångar finns i mattan, så kan du luta dig tillbaka en timme eller så. När du tröttnar kan du skjuta ut samtliga över muren och samtidigt få en otrolig score.

Operation Hormuz
Ett enkelt poke som gör att du får evigt liv.
POKE 25505,234
Starta igen med
SYS 5120

Human Killing Machine
Här kan vi bjuda på oändligt med liv, oändlig energi och motstånd mot bomberna.

Snabbpokes!
Har du en 64/128 kommer här lite pokes som du kan testa. Vad som händer vid de olika pokesen får du själv komma på...

Lazy Jones
POKE 4063,173
SYS 2061

Orpheus
POKE 18870,234
POKE 18871,234
SYS 9728

Manic Miner
POKE 16632,0
POKE 16571,173
SYS 16348

W D W II
POKE 5941,169
POKE 5942,0
POKE 5943,234
SYS 16384

Sentinel
POKE 6679,173
POKE 8512,10
SYS 16128

Trapdoor
POKE 14914,96
SYS 14336

Camelot
POKE 23730,234
POKE 23731,234
POKE 23732,234
SYS 16384

Terra Cognita
POKE 26703,255
SYS 24576

Shadowfire
POKE 25188,173
SYS 16384

Erebus
POKE 2379,106
POKE 2380,9
POKE 2391,106
POKE 2392,9
SYS 12117

Lightforce
POKE 11547,5
SYS 6713

Druid
POKE 35731,12
POKE 35744,0
POKE 37940,0
POKE 39421,0
POKE 35779,76
POKE 35780,215
POKE 35781,139
POKE 39721,255
SYS 5120

1942
POKE 5806,234
POKE 5807,234
SYS 2640

Thrust
POKE 6139,234
POKE 6140,234
POKE 6141,234
SYS 2304

Cauldron II
POKE 40318,65
POKE 40319,208
SYS 32777

Flash Gordon
POKE 22603,234
POKE 22604,169
POKE 22605,0
SYS 12280

Bomb-Jack
POKE 6819,173
SYS 3101

Fairlight
POKE 34413,234
POKE 34414,234
POKE 34420,234
POKE 34421,234
SYS 20992

Forts sid 54

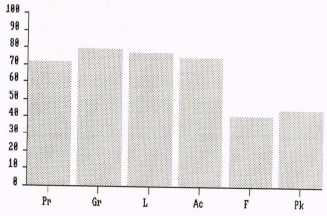
Soft check

VIGILANTE

New York 1994. Polisen törs inte längre patrullera på gatorna. Till och med armén står maktlös mot de gatugäng som tagit över kontrollen av staden. Detta är förutsättningarna i Vigliante från US Gold. Du ska som rättrådig stridsman för lag och ordning ta denna lag i egna händer och ta över där armén misslyckats.

En kvinna har kidnappats av några skinnhuven. Detta kan du ju inte finna dig i, så du ska ut på gatan och slåss med slag och sparkar mot ett större uppbåd av våldsmän. Varje gång du klår en av dem kan du ta dig en liten smula längre mot höger, vilket till slut för dig till slutet av banan där du ska slåss mot en riktigt tuff motståndare.

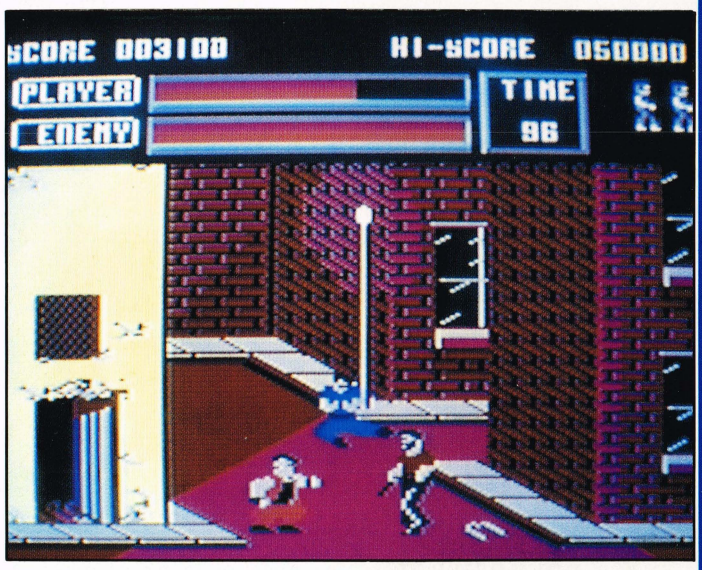
Det finns ett större antal banor i spelet, men vi kom, som kommer att framgå, inte fram till riktigt hur många.



Presentation	**76%**
Grafik	**79%**
Ljud	**78%**
Action	**75%**
Fängslande	**40%**
Pris/kvalitet	**45%**

När man står med asken till spelet Vigilante från US Gold i handen, är det svårt att inte med glädje se fram emot en riktig omgång hänsynslöst datorvåld med massor med grafik- och ljudeffekter. Förpackningen utstrålar nämligen dessa attraktiva signaler i hög grad, men tyvärr är detta också den enda orsaken till att köpa spelet.

Du blir nämligen med till visshet gränsande sannolikhet djupt besviken när du väl laddat in spelet. Visst finns det flotta och väl utförda rörelser, men effektmässigt haltas spelet en hel del efter i jämförelse med alla andra kampspel. Det var först när Hans nästan krupit in i monitorn som vi hörde att det faktiskt kom något som liknade ett ljud av en spark när man träffade motståndaren. Problemet är att det hela tiden spelar en riktigt jobbig hejsan-hoppsan-melodi som totalt lyckas överrösta alla ljudeffekter. Därför får du lätt känslan av att vara en clown i en cirkusmanege som istället för att kasta pajer omkring sig delar ut sparkar till höger och till... ja, vilket andra håll som helst.
Claus



Vigilante är en typisk arkadekonvertering och ser ut som rätt många andra spel till 64:an. Rent grafiskt har US Gold lyckats rätt bra, men det hade varit fint med lite fler olika rörelser.

De första fienderna är verkligt enkla att banka. Enda gången man kan få lite problem är när man blir angripen från två sidor samtidigt. Ett par snabba sparkar betyder dock att man sällan mister så mycket energi på det. När du kommer fram till den stora skurken i slutet är kanpen minst lika given - busen vinner...

I starten trodde jag att det bara var jag som tappat formen efter en månads tågluffande, men trots många försök kom jag aldrig längre än bana 1.

Manualen hjälper heller inte till här, för den förklarar bara hur man laddar in spelet. Det HAR jag kommit på efterhand...

Ljudet är en fantastiskt fin återgivning av en man som slår sig själv i skallen med en påse ris. Detta är ju knappast relevant i ett kampsoel, så jag har lite svårt att hitta en bra orsak för att rekommendera Vigliante.
Hans

FIRST STRIKE

Här kommer Epyx med ett flygsimulationsspel för alla som inte gillar flygsimulatorer. Det krävs nämligen bara ett minimum av insats att lära sig flyga planet, någon manualläsning behövs egentligen inte. Det enda du behöver göra för att få gå upp och pröva dina vingar är att välja ett av de åtta "top secret"-märkta uppdragen.

Första uppdraget är ett träningsd:o där du kan öva dig på mål till lands, till sjöss och givetvis också i luften.

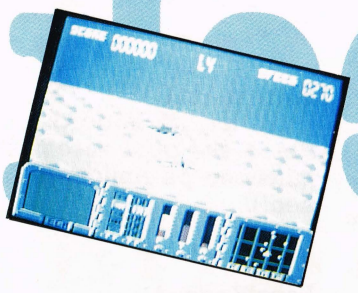
Det första egentliga uppdraget går ut på att bomba en fabrik för kemiska vapen i Libyen. Kom inte och säg att programme-

Vid en första titt på Citadel ringer en liten klocka i skallen och säger "Ding - Paradroid". Både grafisk och handlingsmässigt är det faktiskt en hel del likheter mellan Citadel och denna klassiker. I Citadel ska man bara använda andra robotar som sköld istället för att ta över dem. I starten kan detta vara en smula knepigt, men när du väl genomskådat styrsystemet är det faktiskt rätt enkelt.

Citadel är precis den sort spel som jag gott kan tänka mig att spela länge. Idén är originell (till viss del i varje fall) och det finns både en hel del action och en god portion strategi att roa sig med. Grafik och ljud hänger fint med, även om de olika fienderna gott kan vara lite svåra att se skillnad på.

Så står du och saknar ett spel så att du kan överleva nästa helg utan att dö av tristess, kan jag klart rekommendera Citadel.
Hans

Presentation	**86%**
Grafik	**90%**
Ljud	**85%**
Action	**91%**
Fängslande	**93%**
Pris/kvalitet	**91%**



köpa

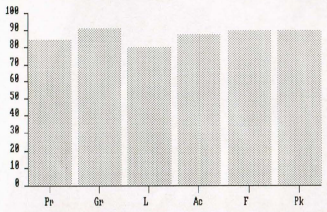
rare inte läser tidningen!

Du börjar med att utrusta ditt plan med bränsle och vapen, du kan lasta upp till 10 ton, och sedan är du luftburen helt automatiskt. Nederst på din skärm finns en liten informationstavla som talar om vilken typ av fiende du kommer att råka på härnäst.

Det spelar ingen roll vilka undanmanövrer du gör i luftstriderna eller om du så ligger i högerkurva hela vägen, kursen är ändå spikrakt mot målet. Det finns flera olika fiendetyper, plan, skepp, helikoptrar, ubåtar, baser, stidsvagnar mm, som ska plockas med olika vapen.

Med mellanrum kommer du fram till din nästa bas på vägen till den kemiska fabriken, och även här går landningen helt av sig själv - till och med den mest härdade actionfreaken kan landa utan vare sig ILS eller VOR radiofyrar.

Presentation	**85%**
Grafik	**90%**
Ljud	**80%**
Action	**87%**
Fängslande	**90%**
Pris/kvalitet	**90%**



Låt det vara sagt med en gång: riktiga män spelar riktiga flygsimulatorer typ Flight Simulator II! Men trots att First Strike är skäligen enkelt i sin uppbyggnad måste jag medge att det faktiskt var riktigt tillfredsställande att få ge herr Khadaffi ett piller från ovan.

Spelet är extremt actionpräglat och det går sällan mer än tre röda sekunder när det inte händer något. Det är speciellt luftstriderna som är kul, även om de inte är ett dugg realistiska. De utspelas alltid på så sätt att fientliga plan lägger sig framför dig så att du utan större problem kan skicka en raket i baken på dem. Dessvärre svarar de igen genom att skjuta baklänges!

Det finns inte många detaljer i landskapet, frånsett de fyrtorn som jag hade den enerverande vanan att braka rätt in i hela tiden i den naiva tron att det var ett skepp. I gengäld är havs- och jordytor snygga, och det finns dessutom ett molntäcke som man kan både flyga i, under och över för att få lite höjdkänsla. Ljudet påminner inte så lite om en gammal Volta med andningsbesvär, men det förmår ändå ge en viss stämning till spelet.

Ett bra stridsspel för alla som gott kan tänka sig att leka jaktplanspilot utan att behöva dras med karta och kompass och som alltid lyckas göra stora hål i jorden i en riktig flygsimulator.

Hans

First Strike-programmerarna har koll på vad de gör. Det sätt som molnen och landskapet far förbi på skärmen är fantastiskt livaktigt. Konceptet är ytterst direkt och det krävs inte många minuters träning innan du är redo att gå igång med de mer avancerade uppdragen.

Målen är mycket allsidiga, men sättet du skiftar vapen på är dessvärre rätt irriterande.

Det är viktigt att poängtera, att First Strike inte har ett dugg med flygsimulatorer att göra. De som skikande rusar iväg när de hör ordet flygsimulator kan sitta kvar i lugn och ro. Det här är istället ett starkt actionorienterat spel som bör tilltala de flesta som gillar att skjuta vilt i hög hastighet.

Claus

CITADEL

På en planet som hittills ansetts vara obebodd, hittar ett rymdforskningsfartyg av en ren slump en ingång till ett teknologiskt avancerat underjordiskt byggnadskomplex. Rasen som på sin tid byggde komplexet är numera utdöd, men komplexets försvarssystem är fortfarande intakt. Det är alltså inte tillrådligt att bara klampa in.

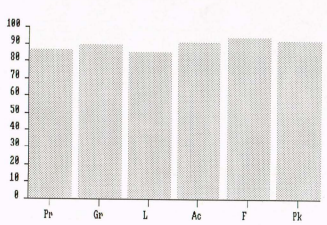
För att kunna utforska detta teknologiska eldorado har du mycket passande utrustats med en monitor som är en slags fjärrstyrd robot. Men hjälp av denna kan du samla in och undersöka intressanta saker du hittar. Du kan också överta kontrollen av vissa av de självstyrande droider du möter, mycket praktiskt eftersom du kan använda dessa som murbräcka och sköld mot försvarssystemets laserkanoner.

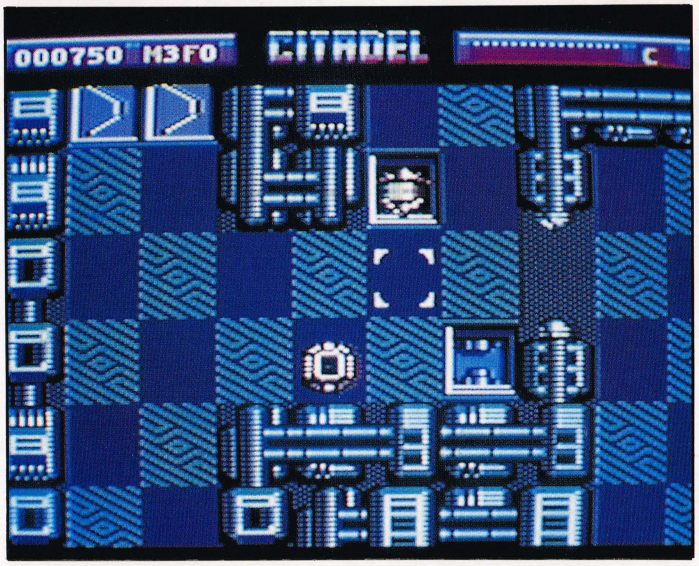
På andra ställen ska du hitta speciella kontakter som slår av kraftfält kring stängda portar.

Eftersom det kostar energi varje gång du blir träffad kan du tacka din skapare för de välplacerade energidepåer som du kan hitta runt omkring.

I komplexets centrum ska du kunna hitta alla den utdöende civilisationens teknologiska läckerheter, men för att komma in dit måste du ta dig igenom 8 olika Citadels som var och en består av 2 våningar.

Motståndet du möter på dessa Citadels blir givetvis värre och värre ju längre in du kommer. Först och främst blir du konfronterad med högre utvecklade droider, dessutom utvecklar sig fällorna i Citadel från att vara förutsägbara och konstanta till att bli slumpmässiga. Du kan därför aldrig veta vad du blir utsatt för när du utlöser en fälla.





Citadel är ett märkvärdit underhållande spel när man väl kommit ordentligt igång med det. När du förstått att utnyttja de droider du fångar kan du systematiskt arbeta dig fram genom Citadellerna. Svårigheten ligger i att banorna blir större och större ju längre in du kommer, därför är det ingen dum idé att börja rita kartor när du väl kommit in ett par Citadeller i spelet.

Grafiken är genomarbetad in i minsta detalj. Ljud? Ja, hektiska bakgrundsschlagers slipper du lyckligtvis att låna dina hörselorgan till i Citadel. Istället har man lagt på en hel del fina ljudeffekter som hjälper till att bygga upp ytterligare en stämningsdimension i spelet.

Citadel är månadens bästa bud på ett rejält fängslande spel, och fängslande även i långa loppet.

Claus

Forts från sid 51

Liv:POKE 43912,44
Energi: POKE 37049,03
Bomber: POKE 37000,168
POKE 37001,131
POKE 37005,208
Startar igen gör du med
SYS 51200

Vendetta

- som faktiskt är ett svenskt spel.
Oändligt med ammunition:
POKE 14191,173
Oändlig energi:
POKE 14466,96
Start igen:
SYS 36864

Giana Sisters

Heja! En tjej, Anna-Lena i Umeå, har skickat in ett tips till de sedermera förbjudna systrarna. Mer brev från tjejer, tack!

När man nått nivå 6 ska man gå fram till bron som leder till den första eldtunnan. I början på den bron ska man hoppa rakt upp, så hamnar man på nivå 9.

Netherworld

Här trycker du helt enkelt på Q, så hoppar du fram en bana. En annan variant är att trycka 2, 4 och E samtidigt.

ARCADE ACTION AID

Starways

Tryck ned 7 efter 5 sekunder, så kommer du fram en bana.

Pacmania

Lite pokes för den lilla datorosten.
POKE 22459,173 slår ifrån spritekollisionen.
POKE 28520,165 ger oändligt antal ostar.

SYS 14336 startar det hela igen.

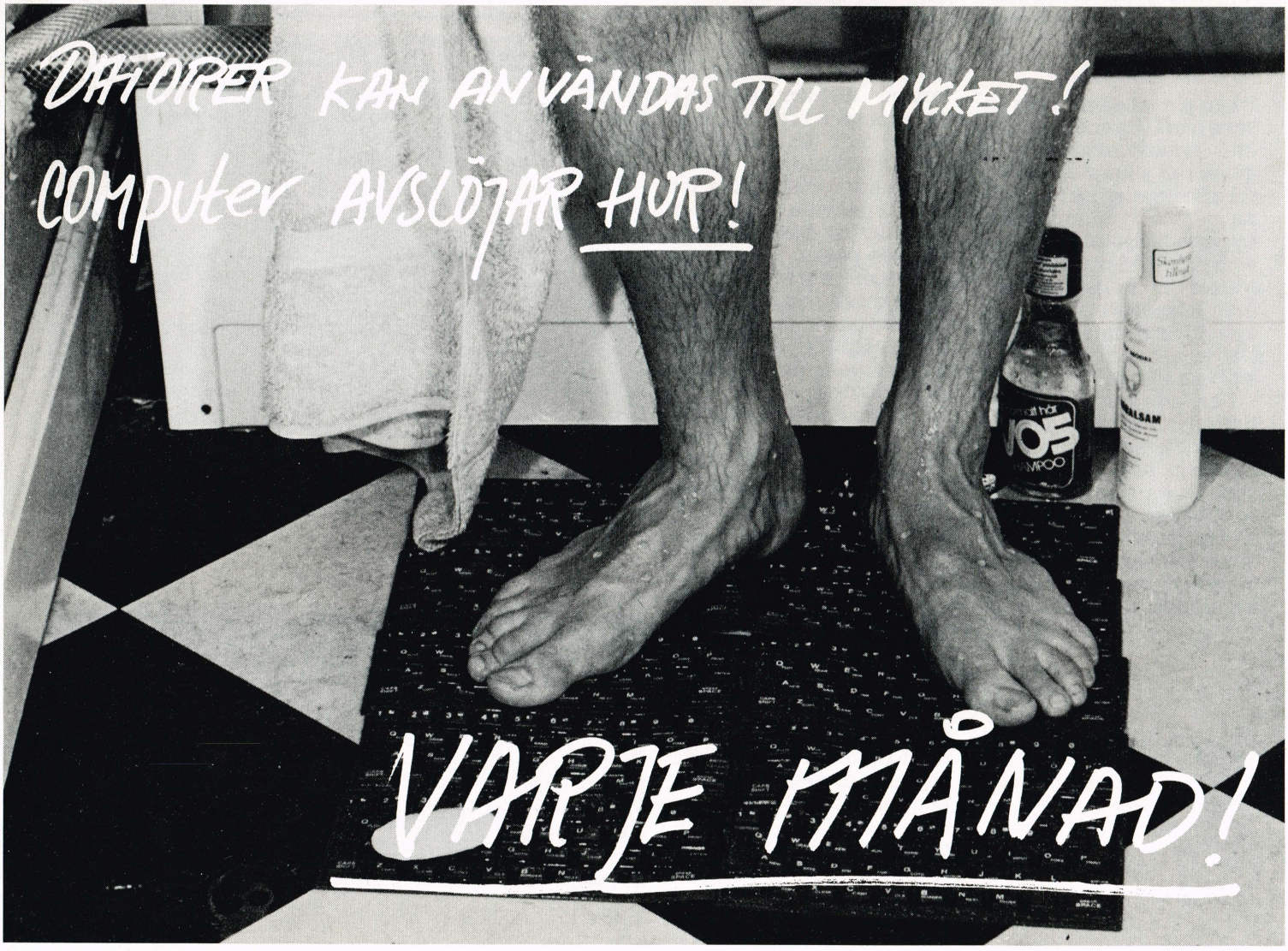
Eliminator

- vill ha lite kodord, och dessa ska göra susen (åtminstone på Amigaversionen!).

LevelKodord
2AMOEBA
3BLOOOP
4CHEEKI
5DOINOK
6ENIGMA
7FLIPME
8GEEGEE
9HANDEL
10ICICLE
11JAMMIN
12KIKONG
13LAPDOG
14MIKADO

Robocop

Trött på att fastna på första nivån? Förtvivla icke! Vänta tills highscoren blinkar och skriv då in SUEDEHEAD, då laddar nivå två in utan att du behöver jobba med första nivån. För tredje nivån är kodordet DISAPPOINTED.



När natten sänker sig...

– säg godnatt till din Amiga eller C64 efter en pixelrik arbetsdag!

COMputer kan nu erbjuda alla våra aktiva läsare ett läckert dammskydd med COMputer-logo till Amiga eller C64.
Vill du undvika damm och smuts i din dator ska du göra följande:

JA TACK, jag vill gärna ha ett dammskydd till Amiga ☐ eller C64 ☐

För lösnummerköpare:

☐ Jag vill köpa ett dammskydd för kr 99:50.
☐ Jag vill prenumerera på COMputer för kr 278:50 och därmed få köpa dammskyddet för endast kr 20, summa kr 298:50.

För prenumeranter:

☐ Jag har skaffat en ny prenumerant och får därför mitt dammskydd gratis!
☐ Jag prenumererar redan och får därmed köpa dammskyddet för kr 79:50.

Jag har skaffat följande nya prenumerant:
Namn:_____
Adress:_____
Postnr.:_____ Ort:_____

Mitt namn och adress:

Namn:_____

Adress:_____

Postnr.:_____ **Ort:**_____

Prenumerationsnummer:_____

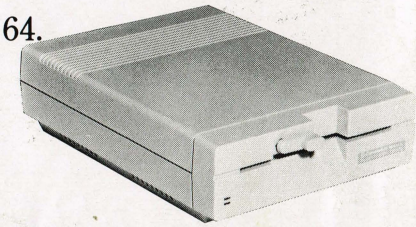
Klipp ut kupongen och skicka den till: Oberoende COMputer
St. Kongensgade 72, DK-1264 Köpenhamn K., Dan

Inte vilka tillbehör som helst

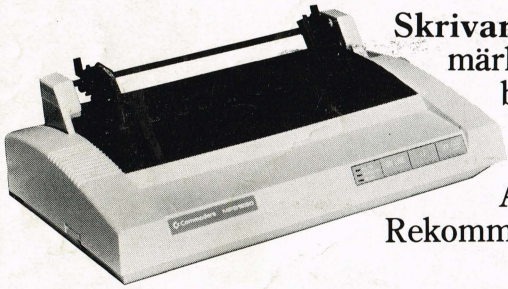
Ställ samma krav på dina tillbehör, som på din dator. Försäkra dig om god funktion och lång livslängd. Och tänk på att det också finns en dag efter köpet.

Vi tar ansvar för våra produkter. De är testade och utvalda för att passa just din Commodore-dator. Och vi ger dig det rikstäckande servicenät, som du kan förvänta dig av marknadens största leverantör av hemdatorer.

Välkommen till närmaste Commodore återförsäljare, så får du veta mera.

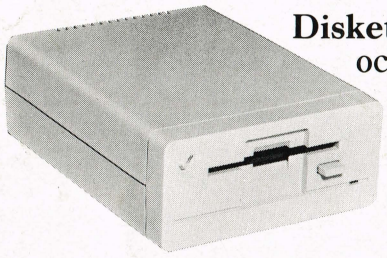


Diskettstation Commodore 1541-II. Ger extra fart åt din Commodore 64. Snabbar upp både sökning av information och laddning av program. 1541 II är avsedd för 5 1/4"-disketter och har en lagringskapacitet på 170 Kb. Passar också Commodore 128 och 128 D.
Rekommenderat ca-pris **2.295:-** inkl moms.

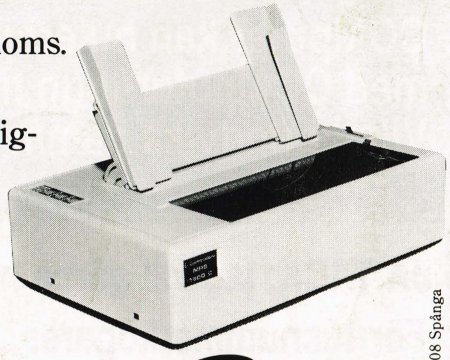


Skrivare Commodore MPS 1250. Skriver snyggt och passar utmärkt för brev och liknande dokument. Många utskriftsmöjligheter, bl a understrykning, fet och kursiv stil. Du kan också välja antalet rader per sida, placering av spalter o s v. Självklart finns både traktor och friktionsmatning. Passar Commodore 64, Amiga och PC.
Rekommenderat ca-pris **2.895:-** inkl moms.

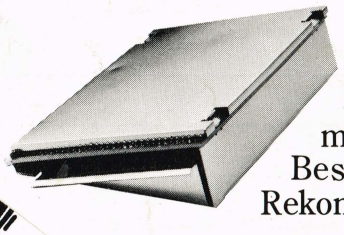
Färgmonitor Commodore A 1084 S. Ger full rättvisa åt din Amiga. Utmärkt färgåtergivning och skarp bild. Matterad bildyta minskar störande reflexer från yttre ljuskällor. Inbyggt ljud. Passar även PC, videobandspelare och TV-turners.
Rekommenderat ca-pris **3.995:-** inkl moms.



Diskettstation Commodore A 1010. För dig som arbetar mycket med Amiga och behöver snabb informationshantering mellan olika program. Diskettstation A 1010 är identisk med Amigans interna diskettstation och arbetar med dubbelsidiga 3.5"-disketter, som har en lagringskapacitet på 880 Kb.
Rekommenderat ca-pris **1.995:-** inkl moms.



Skrivare Commodore MPS 1500 C. En skrivare med stora möjligheter trots lågt pris. Kan användas för utskrift av både färggrafik och text. Många inställningsmöjligheter, exempelvis olika teckenavstånd, normal och fet stil, understrykning, index- och exponentfunktion. Avsedd för Commodore Amiga.
Rekommenderat ca-pris **3.995:-** inkl moms.



Minnesexpansion Commodore A 501. Utvidgar arbetsminnet i din Amiga 500 från 512 Kb till 1 Mb. Ett nästan nödvändigt tillbehör med tanke på de allt kraftfullare programmen. Bestyckad med klocka och batteri-backup.
Rekommenderat ca-pris **1.895:-** inkl moms.

Commodore

Commodore AB, Box 8184, 163 08 Spånga